

TUGAS AKHIR - TF 141581

RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)

MOHAMMAD MUSA BACHROWI NRP 02311440000089

Dosen Pembimbing Dr.Ir. Ronny Dwi Noriyati, M.Kes

DEPARTEMEN TEKNIK FISIKA Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya 2018



FINAL PROJECT - TF 141581

DESIGN STAND ALONE PV CHARGING SYSTEM BASED ON SOLAR TRACKER USING FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO) CONTROL

MOHAMMAD MUSA BACHROWI NRP 023114140000089

Supervisor Dr.Ir. Ronny Dwi Noriyati, M.Kes

DEPARTMENT OF ENGINEERING PHYSICS Faculty of Industrial Technology Institut Teknologi Sepuluh Nopember Surabaya 2018

Halaman ini sengaja dikosongkan

PERNYATAAN BEBAS PLAGIRISME

Saya yang bertanda tangan di bawah ini:

Nama : Mohammad Musa Bachrowi

NRP : 02311440000089

Departemen : Teknik Fisika FTI-ITS

Dengan ini menyatakan bahwa tugas akhir saya berjudul RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO) adalah bebas dari plagiasi. Apabila pernyataan ini terbukti tidak benar, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenarbenarnya.

Surabaya, 1 Juni 2018

Yang membuat pernyataan,

Mohammad Musa Bachrowi

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN TUGAS AKHIR

RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)

Oleh : <u>Mohammad Musa Bachrowi</u> NRP :02311440000089

> Surabaya, 1 Juni 2018 Mengetahui

Dosen Pembimbing

haut

Dr.Ir. Ronny Dwi Noriyati, M.Kes NIPN. 19571126 198403 2 002



Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN TUGAS AKHIR

RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)

Diajukan Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana Teknik pada Bidang Studi Rekayasa Instrumentasi dan Kontrol Program Studi S-1 Departemen Teknik Fisika Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember

Oleh : <u>MOHAMMAD MUSA BACHROWI</u> NRP.0231140000089

Disetujui oleh Tim Penguji Tugas Akhir :

1.	Dr.Ir. Ronny Dwi Noriyati, M.Kes(Pembimbing 1)
2.	Dr. Ir. Ali Musyafa', M.Sc
3.	Dr. Imam Abadi, ST, M.T(Penguji 1)
4.	Detak Yan Pratama, S.T, M.Sc(Penguji 2)

SURABAYA JUNI, 2018

Halaman ini sengaja dikosongkan

RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)

Nama Mahasiswa	: Mohammad Musa Bachrowi
NRP	: 02311440000089
Departemen	: Teknik Fisika FTI-ITS
Dosen Pembimbing	: Dr.Ir. Ronny Dwi Noriyati, Mkes

Abstrak

Membangun suatu pembangkit listrik tenaga surya memerlukan investasi awal yang cukup mahal. Sehingga perlu dirancang komponen pembangkit listik tenaga surya yang mampu meningkatkan performansi panel surya. Salah satu untuk meningkatkan performansi cara panel surva menggunakan solar tracker. Penggunaan solar tracker juga membutuhkan konsumsi energi tambahan untuk mengarahkan panel surya tegak lurus matahari. Sehingga digunakan baterai sebagai suplai energi tambahan tersebut. Energi vang dihasilkan Panel surva dapat disimpan di baterai dengan menggunakan bantuan solar charge controller. Solar charge controller yang handal, optimal, efisien dan ekonomis akan meningkatkan performansi penyimpanan energi panel surya. Pada penelitian ini konverter DC-DC digunakan adalah tipe Single Ended Primary Inductance Converter (SEPIC). Kontrol tegangan pengisisan baterai menggunakan Fuzzy-LDWPSO. Diketahui dari simulasi konrol Fuzzy LDWPSO meningkatkan nilai RiseTime 0.0021 detik, SettlingTime 0.0135 detik PeakTime 0.0056 detik, eror steady state 3 % dan merunkan Maximum Overshoot 27,5%. Pengujian pada hardware dan simulasi dilakukan untuk mengetahui pengaruh duty cycle dan frekuensi pensaklaran pada konverter SEPIC. Semakin besar nilai duty cycle semakin besar tegangan output yang dihasilkan konverter. Nilai suhu dan irradiasi juga berpengaruh terhadap tegangan input dan arus input yang masuk pada konverter. Kenaikan arus pengisian pada PV kondisi fix meningkat 36,15% selama 8,5 jam.

Kata kunci : Solar Tracker, Konverter SEPIC, Linear Decreasing Weight Particle Swarm Optimization, duty cycle.

DESIGN STAND ALONE PV CHARGING SYSTEM BASED ON SOLAR TRACKER USING FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO) CONTROL

Name	: Mohammad Musa Bachrowi
NRP	: 0231144000089
Department	: Engineering Physics FTI-ITS
Supervisor	: Dr.Ir. Ronny Dwi Noriyati, Mkes

Abstract

Building a solar power plant requires an expensive initial investment. It is necessary to design solar pan power generating components that can improve the performance of solar panels. One of the ways to improve the performance of solar panels using solar tracker. Solar trackers also requires the consumption of additional energy to direct solar panels perpendicular to the sun. So use the battery as an additional energy supply. The energy produced Solar panels can be stored in batteries by using the help of solar charge controller. Solar charge controller that is reliable, optimal, efficient and economical will improve the energy storage performance of solar panels. In this study DC-DC converter used is Single Ended Primary Inductance Converter (SEPIC) type. Battery charge voltage control using Fuzzy-LDWPSO. Known from Fuzzy LDWPSO control simulation increases RiseTime value 0.0021 seconds, SettlingTime 0.0135 seconds PeakTime 0.0056 sec. steady state errors 3% and Maximum Overshoot 27.5%. Tests on hardware and simulations are performed to determine the effect of duty cycle and switching frequency on SEPIC converters. The greater the duty cycle value the greater the output voltage generated by the converter. Temperature and irradiation values also affect the input voltage and input current in the converter. The increase of charging current in PV fix condition increased 36.15% for 8.5 hours.

Keywords: Solar Tracker, SEPIC Converter, Linear Decreasing Weight Particle Swarm Optimization, duty cycle.

KATA PENGANTAR

Alhamdulillahirobbil'alamin, segala puji bagi Allah SWT atas berkat, rahmat dan hidayah-Nya sehingga penulis diberikan kesehatan, kemudahan, dan kelancaran dalam menyusun laporan tugas akhir yang berjudul:

"RANCANG BANGUN STAND ALONE PV CHARGING SYSTEM BERBASIS SOLAR TRACKER MENGGUNAKAN KONTROL FUZZY MODIFIED PARTICLE SWARM OPTIMIZATION (MPSO)"

Tugas akhir ini merupakan salah satu persyaratan akademik yang harus dipenuhi dalam Program Studi S-1 Teknik Fisika FTI-ITS. Penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

- 1. Agus Muhamad Hatta, S.T., M.Si., Ph.D. selaku kepala Departemen Teknik Fisika-ITS.
- 2. Ir. Ronny Dwi Noriyati, Mkes selaku dosen pembimbing tugas akhir ini, yang selalu memberikan bimbingan dan semangat pada penulis.
- 3. Segenap Bapak/Ibu dosen pengajar di DepartemenTeknik Fisika ITS.
- 4. Mohammad Wardi dan Siti Rofiah selaku orang tua penulis serta segenap keluarga besar penulis yang selalu memberikan doa dan dukungan untuk penulis.
- 5. Rekan-rekan Departemen Teknik Fisika ITS dan rekanrekan seperjuangan TA yang senantiasa memberikan motivasi dan bantuan dalam penyelesaian laporan tugas akhir ini.

Penulis menyadari bahwa mungkin masih ada kekurangan dalam laporan ini, sehingga kritik dan saran penulis terima. Semoga laporan ini dapat berguna dan bermanfaat bagi penulis dan pihak yang membacanya.

Surabaya, 1 Juni 2018

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

HAI	LAMAN JUDULi		
PER	NYATAAN BEBAS PLAGIARISMEv		
LEN	/IBAR PENGESAHANError! Bookmark not defined.		
ABS	STRAKError! Bookmark not defined.		
ABS	STRACTError! Bookmark not defined.		
KA	ГА PENGANTARxv		
DAI	FTAR ISI xvii		
DAI	FTAR GAMBARxix		
DAI	FTAR TABELxxi		
BAI	31		
1.1	Latar Belakang 1		
1.2	Rumusan Masalah3		
1.3	Tujuan		
1.4	Batasan Masalah3		
BAI	3 II		
2.1	Panel Surya5		
2.2	Solar Tracker (Sistem Penjejak Matahari)7		
2.3	Konverter SEPIC		
2.4	Logika Fuzzy10		
2.5	Charging Mode14		
2.6	Modified Particle Swarm Optimization15		
BAI	3 III		
3.1	Diagram Alir17		
3.2	Spesifikasi sistem		
3.3	Pengambilan Data Primer		
3.4	Perancangan kontrol Fuzzy-MPSO		
3.5	Perancangan Hardware		
BAB IV			
4.1	Simulasi konverter SEPIC pada ISIS41		
4.2	Pengujian Karakteristik konverter SEPIC		

4.3	Simulasi kontrol Penjejakan Set Point Fuzzy dan		
	Fuzzy-LDWPSO	.46	
4.4	Simulasi Uji Tracking Fuzzy dan Fuzzy-LDWPS	SO	
	dengan Variasi Kondisi Klimatik	.48	
4.5	Simulasi Charging kontrol Fuzzy dan		
	Fuzzy LDWPSO	.51	
4.6	Pengujian hardware dengan kontrol		
	Fuzzy LDWPSO	.52	
4.7	Pengujian Hardware dengan PV kondisi		
	Fix dan tracking	.53	
BA	BV		
5.1	Kesimpulan	.57	
5.2	Saran	.57	
DA	FTAR PUSTAKA	.57	

DAFTAR GAMBAR

Gambar 2.1 Kurva I-V Panel Surya ^[1]	7
Gambar 2.2 single axis solar tracking ^[1]	8
Gambar 2.3 two axis solar tracking ^[1]	8
Gambar 2.4 Rangkaian konverter SEPIC ^[7]	9
Gambar 2.5 Sistematika logika Fuzzy ^[8]	11
Gambar 2.6 Fungsi keanggotaan segitiga	14
Gambar 2.7 Fungsi keanggotaan trapesium	14
Gambar 2.8 Fungsi keanggotaan gaussian	14
Gambar 2.9 Fungsi keanggotaan generalized bell	14
Gambar 2.10 Diagram 3 fasa charging	14
Gambar 2.11 Pseudo code Particle Swarm Optimization	
(PSO) secara umum ^[9]	16
Gambar 3.1 Diagram alir penelitian	17
Gambar 3.2 Diagram blok sistem secara garis besar	19
Gambar 3.3 Kurva karakteristik panel surya	20
Gambar 3.4 Rangkaian Single Ended Primary	
Inductance Converter	21
Gambar 3.5 Pemodelan SEPIC pada PSIM	22
Gambar 3.6 Karakteristik Arus yang melewati (1)C1	
(2)L1 (3)Cs (4) L2 (5)R	22
Gambar 3.7 Karakteristik tegangan input (Merah),	
Tegangan Output (biru)	22
Gambar 3.8 Diagram blok kontrol Fuzzy MPSO	26
Gambar 3.9 Fungsi keanggotaan input Eror dan delta	
eror Tegangan	26
Gambar 3.10 Fungsi keanggotaan output Duty Cycle	27
Gambar 3.11 Langkah optimasi PSO	29
Gambar 3.12 fungsi keanggotaan fuzzy-LDWPSO	33
Gambar 3.13 Hardware konverter SEPIC	35
Gambar 3.14 Pengukuran nilai induktor	36
Gambar 3.15 Sinyal PWM dari arduino	36
Gambar 3.16 Rangkaian penguat sinyal PWM TLP250	36
Gambar 3.17 Sinyal PWM yang telah dikuatkan oleh	
TLP250	37

Gambar 3.18 Pengukuran resistor	37
Gambar 3.19 Arduino uno	37
Gambar 3.20 Sensor Tegangan	37
Gambar 4.1 Setup Simulasi SEPIC pada software	
ISIS9.0	41
Gambar 4.2 Setup alat SEPIC	41
Gambar 4.3 Grafik perbedaan Tegangan dan arus	
simulasi dan alat	42
Gambar 4.4 Pengaruh Pengaruh Duty cycle terhadap	
tegangan output konverter	45
Gambar 4.5 Tracking tegangan 14.55 volt pada kondisi	
STC	47
Gambar 4.6 Variasi dara Irradiasi (W/m ²)	48
Gambar 4.7 Variasi data suhu (°C)	49
Gambar 4.8 Hasil Uji Tracking Fuzzy dan Fuzzy-	
LDWPSO dengan Variasi Kondisi Klimatik	50
Gambar 4. 9 Ripple arus pengisisan baterai kontrol fuzzy	
dan Fuzzy LDWPSO	51
Gambar 4.10 Arus pengisisan kontrol fuzzy-LDWPSO	
hardware	52
Gambar 4.11 Set Up pengujian PV fix dan PVTracking	52
Gambar 4.12 Perbandingan arus pengisian Fuzzy	
LDWPSO Fix, tracker, SCC fix dan tracker	51

DAFTAR TABEL

Tabel 2.1	Perbandingan konverter SEPIC dengan	
	berberapa konverter lain	10
Tabel 3.1	Spesifikasi Panel Surya yang digunakan	20
Tabel 3.2	parameter Komponen konverter SEPIC	
	yang digunakan	21
Tabel 3.3	Data Irradiasi, Suhu, Arus dan Tegangan	
	Panel Surya	24
Tabel 3.4	Rule Base Fuzzy	28
Tabel 3.5	Inisialisasi awal posisi dan kecepatan	
	partikel	24
Tabel 3.6	Hasil perhitungan MSE tiap partikel	30
Tabel 3.7	Hasil Update posisi dan kecepatan	
	partikel	31
Tabel 4.1	Data tegangan dan arus simulasi dan alat	42
Tabel 4.2	Pengaruh Frekuensi switching terhadap	
	tegangan output konverter	43
Tabel 4.3	Pengaruh Duty cycle terhadap tegangan	
	output konverter	44
Tabel 4.4	Perbandingan parameter kontrol fuzzy	
	dan fuzzy LDWPSO	54
Tabel 4.5	Pengujian kondisi PV Fix dan Tracking	54

Halaman ini sengaja dikosongkan

BAB I PENDAHULUAN

1.1 Latar Belakang

Energi merupakan salah satu kebutuhan utama dalam kehidupan manusia. Peningkatan kebutuhan energi dapat merupakan indikator peningkatan kemakmuran, namun bersamaan dengan itu juga menimbulkan masalah dalam usaha penyediaannya. Seiring dengan bertambahnya jumlah penduduk jumlah energi yang digunakan juga meningkat. Konsumsi energi yang besar mengakibatkan ketersediaan energi semakin menipis. Konsumsi energi listrik di Indonesia mencapai 202.845,82 GWh, merupakan nominal yang cukup besar juga dibandingkan beberapa negara di Asia Tenggara lainnya.

Terkait dengan pembangkit listrik tenaga surya, Indonesia sebagai negara tropis mempunyai potensi energi surya yang cukup besar. Potensi energi matahari di Indonesia berkisar 4,8 kWh per meter persegi hari. Untuk memanfaatkan potensi energi surya tersebut, ada 2 (dua) jenis teknologi yang sudah diterapkan yaitu teknologi energi surya termal dan energi panel surya. Energi panel surya digunakan untuk memenuhi kebutuhan listrik, pompa air, televisi, telekomunikasi dan lemari pendingin di wilayah terpencil. Energi listrik alternatif yang dihasilkan panel surya biasanya disimpan terlebih dahulu pada sistem penyimpanan (baterai) sebelum digunakan. Panel surya menjadi sumber energi terbarukan yang menawarkan banyak keuntungan diantaranya Energi yang digunakan berasal dari matahari, tidak menghasilkan polusi, biaya perawatan rendah dan tidak menghasilkan noise

Konversi solar Energi menjadi tenaga listrik juga bergantung pada sudut penerimaan radiasi matahari. Ada dua jenis penggunaan panel yaitu bentuk tetap dan pelacak

matahari (solar tracker). Dewasa ini telah banyak pengoptimalan dilakukan pada panel surya dengan sistem solar tracking. Solar tracking atau biasa disebut sistem penjejak matahari salah satu manfaatnya adalah untuk mengoptimalkan pengumpulan energi matahari. Terdapat dua macam penjejak matahari berdasarkan sumbunya sistemnya vaitu subu tunggal dan sumbu ganda. Efisiensi sistem penjejak matahari sumbu ganda diketahui 81,68% lebih tinggi dari pada panel tetap sedangkan efisiensi pelacak sumbu tunggal hanya 32,17% lebih tinggi dari pada panel tetap ^[1]. Pokok pada penelitian solar tracking difokuskan pada penambahan algoritma Maximum Power Point Tracking (MPPT) pada sistem panel PV. MPPT merupakan konsep yang membantu menghasilkan daya maksimal yang dapat dicapai oleh PV. Daya keluaran PV dipengaruhi oleh tingkat irradiasi matahari dan suhu yang diterima permukaan panel PV. Jika irradiasi yang diterima oleh panel PV semakin tinggi maka arus yang dihasilkan akan semakin tinggi, namun jika suhu panel terlalu tinggi tegangan yang didapat akan semakin rendah. Sudah banyak penelitian mengenai MPPT yang salah satunya menggunakan Fuzzy Particle swarm Optimization(PSO). Dari hasil eksperimen dan simulasi diketahui waktu pelacakan dan akurasi meningkat sebesar 0,88% dan 0,93 % dari Fuzzy tanpa PSO^[2]

Pembangunan pembangkit listrik tenaga surya membutuhkan perencanaan yang sangat baik agar tidak menimbulkan dampak yang berlebihan bagi lingkungan. Untuk membangun suatu pembangkit listrik tenaga surya memerlukan investasi awal yang cukup mahal. Sehingga perlu dirancang komponen pembangkit listik tenaga surya. Salah satunya solar charge controller yang handal, optimal, efisien dan ekonomis sehingga diperoleh perencanaan pembangunan PLTS yang efisien dan optimal. Dalam penelitian sebelumnya tugas akhir ini, memaparkan rancang bangun dari *charging system controller* dimana set point *charge* akan menyesuaikan dengan keadaan tegangan baterai sehingga kontroler ini akan bekerja dengan efisien dan optimal untuk mendukung perencanaan pembangunan PLTS yang baik. Oleh karena pada tugas akhir ini akan dibuat sebuah *charging system* dengan judul "Rancang Bangun *Stand Alone* PV *Charging System* Berbasis *Solar Tracker* Menggunakan Kontrol Fuzzy *Modified Particle Swarm Optimization* (MPSO)".

1.2 Rumusan Masalah

Dari latar belakang diatas, permasalahan yang diangkat dalam penelitian tugas akhir ini adalah sebagai berikut:

- 1. Bagaimana menentukan parameter pada sistem kontrol fuzzy MPSO
- 2. Bagaimana menentukan parameter charging system
- 3. Bagaimana peningkatan perfomansi kinerja *charging system* dengan metode kontrol fuzzy MPSO baik dari sisi simulasi dan real plant.

1.3 Tujuan

Tujuan utama dari rancang bangun alat ini adalah:

- 1. Mengimplementasikan kontrol fuzzy MPSO pada *Charging system solar tracker*
- 2. Mengetahui performansi solar charge controller dengan metode kontrol fuzzy MPSO.

1.4 Batasan Masalah

Adapun batasan lingkup permasalahan dari penelitian tugas akhir ini yaitu:

- 1. Tipe konverter yang digunakan adah konvereter SEPIC (*Single Ended Primary Inductance Converter*).
- 2. Seluruh konsumsi kebutuhan energi internal akan di support oleh Baterai.
- 3. Spesifikasi PV panel yang digunakan adalah 250WP.
- 4. Kapasitas baterai yang digunakan adalah VRLA 70Ah.
- 5. Metode penalaran Fuzzy menggunakan Fuzzy Mamdani.

- 6. Modified PSO yang digunakan adalah Linear Decreasing Weight Particle Swarm Optimization (LDWPSO).
- 7. Lokasi pengujian di Departement Teknik Fisika ITS Surabaya.

BAB II TINJAUAN PUSTAKA

2.1 Panel Surya

Photovoltaic (PV) atau panel surva merupakan suatu piranti yang berfungsi untuk menkonversi energi cahaya matahari menjadi energi listrik. Material penyusun PV terbuat dari semikonduktor. Terdapat dua tipe semikonduktor dalam solar cell, yaitu tipe-p dan tipe-n. Semikonduktor type-p memiliki kekurangan ion negatif (negative acceptor ions) sehimgga memiliki banyak hole, sedangkan semikonduktor tipe-n mempunyai kelebihan ion negatif (positive donor ions) sehingga memiliki banyak elektron bebas. Gabungan kedua tipe semikonduktor ini akan menimbulkan kontak permukaan yang disebut dengan p-n junction. P-n Juction membuat susunan elektron keduanya menjadi teratur. Elektron-elektron berlebih yang terdapat pada pita-n akan segera mengisi lubang-lubang elektron (hole) yang terdapat pada pita-p. Kondisi ketika cahaya matahari mengenai permukaan semikonduktor akan membuat elektron-elektron yang terdapat dalam pita energi mempunyai energi yang cukup untuk lepas dari ikatan, sehingga elektron akan mengalir kedalam circuit untuk menuju beban (tereksitasi). Elektron yang tereksitasi tersebut akan meninggalkan lubang yang nantinya akan diisi oleh elektron yang lainnya.

Parameter dari panel surya dapat diperoleh melalui karakteristik arus-tegangan antara lain arus hubungan singkat (Isc) tegangan rangkaian terbuka (Voc), daya maksimum dan *fill factor*^[3]. Tegangan pada panel surya dipengaruhi oleh suhu permukaan panel. Arus yang diahasilkan oleh panel surya dipengaruhi oleh nilai irradiasi matahari yang diterima permukaan panel. Persamaan (2.1) menunjukkan hubungan pengaruh suhu permukaan panel terhadap tegangan yang dihasilkan oleh PV

5

$$V_{Pv} = V_{stc} \times f_{temp} \times \left(\frac{lr}{lr_{STC}}\right)$$
(2.1)

Dimana

$$F_{\text{temp}} = 1 + \alpha_p \left(Tc - Tc_{STC} \right)$$
(2.2)

Keterangan:

 $V_{PV} = Tegangan PV (V)$

- V_{STC}= Tegangan PV saat Standart Temperature Condition (V)
- Ir = Irradiasi matahari pada kondisi uji (W/m^2)
- $Ir_{STC} = Irradiasi matahari pada kondisi standart$ Temperature Condition (W/m²)
- $F_{temp} = Faktor susut akibat pengaruh suhu (%)$
- α_p = Koefisien temperatur tegangan (%/°C)
- Tc = Temperatur PV pada kondisi uji ($^{\circ}$ C)
- Tc_{STC}= Temperatur PV pada *standart Temperature* Condition (°C)

Sedangkan arus yang dihasilkan oleh panel surya lebih dipngaruhi oleh paparan irradiasi yang mengenai permukaaan panel^[4]

$$I_{Pv} = (I_{STC} + K\Delta_T) \left(\frac{lr}{lr_{STC}}\right)$$
(2.3)

Keterangan:

 $I_{PV} = arus PV (A)$

- I_{STC}= Arus PV pada kondisi *standart Temperature* Condition (A)
- $K = Konstanta boltzman (1,38x10^{-23} m^2 Kg/s^2C)$
- $\Delta_{\rm T}$ = Perbedaan suhu (C)

Panel surya adalah perangkat rakitan sel-sel potovoltaik yang mengkonversi sinar matahari menjadi listrik. Arus dan tegangan memiliki spesifikasi sesuai dengan daya maksimum yang dapat dihasilkan oleh panel surya. Gambar 2.1 merupakan hubungan arus dan tegangan yang dapat dihasilkan oleh panel surya.



Gambar 2.1 Kurva I-V Panel Surya^[1]

Pembangkitan arus pada sel surya memiliki beberapa proses, diantaranya sebagai berikut :

- 1. Cahaya dalam bentuk partikel foton jatuh pada permukaan sel surya, lalu diserap dan menghasilkan *electron* dan *hole*. Elektron pada semikonduktor tipe p dan *hole* pada tipe n yang dihasilkan tidak stabil.
- 2. Untuk mencegah rekombinasi digunakan p-n *junction* yang memisahkan *electron* dengan *hole*.

Posisi ideal panel surya adalah menghadap langsung ke sinar matahari (untuk memastikan efisiensi maksimum). Panel surya modern memiliki perlindungan *overheating* yang baik dalam bentuk semen konduktif termal. Perlindungan *overheating* sangat penting karena panel surya mengkonversi kurang dari 20% dari energi surya yang ada menjadi listrik, sementara sisanya akan terbuang sebagai panas dan tanpa perlindungan yang memadai kejadian *overheating* dapat menurunkan efisiensi panel surya secara signifikan^[1].

2.2 Solar Tracker (Sistem Penjejak Matahari)

Gerak rotasi bumi menyebabakan pergerakan matahari dari terbit hingga terbenam. Sedangkan panel surya (PV) sangat bergantung pada irradiasi matahari. Irradiasi matahari akan maksimal jika panel surya menghadap tegak lurus terhadap arah matahari. Sistem Penjejak Matahari berfungsi mengendalikan orientasi panel surya (PV) agar tegak lurus terhadap matahari.



Gambar 2.3 : two axis solar tracking^[1]

Terdapat dua jenis Sistem penjejakan Matahari, yaitu sistem penjejakan aktif dan sistem penjejakan pasif. Sistem penjejakan dikatakan aktif jika sistem penjejakannya melakukan pengukuran langsung terhadap besaran fisis objek misalnya irradiasi matahari. Sistem penjejakan dikatakan pasif jika sistem penjejakan dilakaukan berdasarkan perhitungn astronomi. Berdasarkan jumlah sumbu rotasinya sistem tracking dibagi menjadi dua, yaitu *two axis* dan *single axis*. *Two axis solar tracking* memiliki dua sumbu putar untuk menjejak matahari, yaitu sumbu *pitch* dan *yaw*. Sumbu *pitch* berfungsi untuk melakukan rotasi dari utara ke selatan atau sebaliknya. Sumbu *yaw* berfungsi untuk melakukan rotasi dari timur ke barat atau sebaliknya.

2.3 Konverter SEPIC

SEPIC (Single Ended Primary Inductance Converter) topologi DC/DC adalah salah satu konverter vang menghasilkan tegangan positif dari tegangan masukan positif (non Inferting). Prinsip kerja SEPIC dapat menurunkan atau menaikan tegangan output layaknya prinsip kerja Buck-Boost konverter. SEPIC terdiri dari Dua buah inductor, dua buah kapasitor, sebuah dioda dan MOSFET sebagai saklar dan rangkaian pengontrol (Driver) untuk mengatur MOSFET. Pada rangkaian pengontrol nilai dari duty cycle diatur agar tegangan output konverter sesuai dengan tegangan yang diinginkan.



Gambar 2.4 : Rangkaian konverter SEPIC^[5]

Secara garis besar kinerja dari SEPIC dibagi menjadi 2 yaitu:

1. MOSFET ON : pada posisi ini MOSFET berlaku sebagai penghubung singkat. Dioda akan OFF dan positif sumber terputus dengan posistif beban

2. MOSFET OFF : pada posisi ini MOSFET berlaku sebagai hubungan terbuka (rangkaian terbuka). Dioda akan ON sehingga positif sumber terhubung dengan positif beban. Jika duty cycle diperkecil maka arusyang melalui positif beban lebih besar

Berikut perbandingan konverter sepic dibandingkan beberapa konverter lainnya: ^[6]

Feature	Buck- Boost	Cuk	Positive Buck- boost	SEPIC	Flyback
Output Voltage polarity	Invert	Invert	Non-Invert	Non-Invert	Non-Invert
Input Current	Pulsating	Nonpusating	Depends of operation mode	Non- Pulsating	Pulsating
Switch Drive	Floated	Floated	One Floated One grounded	Grounded	Grounded
Efficiency	Low	Medium	High with only one stage is acive	Medium	Low
Cost	Medium due to float drive	Medium due to additional block capacitor	High due to an additional switch and diode, a more complex drive circuit	Medium due to additional block capacitor	Low due to grounded switch and so block capacitor

 Tabel 2.1 : Perbandingan konverter SEPIC dengan berberapa konverter lain^[7]

2.4 Logika Fuzzy

Logika Fuzzy merupakan pengembangan dari logika Boolean yang menitik beratkan pada konsep *kebenaran sebagian*. Saat logika klasik (boolean) menyatakan bahwa segala hal dapat diekspresikan dalam istilah biner (0 atau 1, hitam atau putih, ya atau tidak), logika fuzzy menggantikan kebenaran boolean dengan tingkat kebenaran. Sistem logika fuzzy pertama kali diperkenalkan oleh Prof. Lofti A. Zadeh pada tahun 1965. Logika fuzzy adalah cabang dari sistem kecerdasan buatan (artificial intelligence), yaitu sebuah logika yang dikembangkan berdasarkan metode dan prinsip dasar berpikir penalaran manusia. Berikut adalah sistematika dasar dari logika fuzzy :



1. Input

Input logika fuzzy berupa bilangan crisp

2. Fuzzyfication

Kolom fuzzifikasi berguna untuk mengkonversi bilangan crisp menjadi suatu bilangan fuzzy (variable linguistik) menggunakan operator fuzzifikasi. Proses fuzzifikasi merupakan ciri utama dalam penalaran terdapat fungsi keanggotaan logika fuzzy. karena fuzzy (membership function) yang berfungsi untuk menentukan letak keberadaan elemen dalam suatu himpunan fuzzy.

- 3. Rule Base & Inference System aturan-aturan dasar logika fuzzy yang Berisikan dimuat dalam aturan IF...THEN.
- 4. Defuzzyfication

Keputusan yang dihasilkan dari proses penalaran masih dalam bentuk fuzzy, yaitu berupa derajat keanggotaan keluaran. Hasil ini harus diubah kembali menjadi bilangan crisp melalui proses defuzzyfikasi.

5. Output

Keluaran pada logika fuzzy berupa bilangan crisp.

Sebuah himpunan fuzzy dapat dikarakterisasikan seluruhnya melalui fungsi keanggotaaanya. Fungsi keanggotaan pada satu dimensi atau single-input dapat direpresentasikan ke dalam 4 bentuk, yaitu fungsi keanggotaan segitiga, fungsi keanggotaan trapesium, fungsi keanggotaan Gaussian, dan fungsi keanggotaan generalized Bell.

a. Fungsi Keanggotaan Segitiga

Fungsi keanggotaan segitiga dibentuk oleh 3 parameter {a, b, c} yang dapat dideskripsikan oleh Persamaan (2.4)



Gambar 2.6 fungsi keanggotaan $\mu[x] = \max\left(\min\left(\frac{(x-a)}{(b-a)}, \frac{(c-x)}{(c-b)}\right), 0\right)$

b. Fungsi Keanggotaan Trapesium

Fungsi keanggotaan trapesium dibentuk oleh 4 parameter {a, b, c, d} yang dapat dideskripsikan oleh Persamaan (2.5)

(2.4)



Gambar 2.7 fungsi keanggotaan trapesium

$$\mu[x] = \max\left(\min\left(\frac{(x-a)}{(b-a)}, 1, \frac{(d-x)}{(d-c)}\right), 0\right)$$
(2.5)

c. Fungsi keanggtaan gaussian

keanggotaan Fungsi Gaussian dibentuk oleh 2 $\{c, \sigma\}$. c merepresentasikan pusat parameter fungsi keanggotaan dan σ merepresetasikan lebar ruang fungsi keanggotaan. Fungsi keanggotaan Gaussian dapat dideskripsikan oleh Persamaan (2.6)

$$\mu[x] = e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2}$$
(2.6)



d. Fungsi keanggotaan generalized bell

Fungsi keanggotaan generalized bell dibentuk oleh 3 parameter {a, b, c} dimana nilai b bernilai positif dan nilai b negatif mengakibatkan kurva terbuka ke atas. Fungsi keanggotaan generalized bell dapat dideskripsikan oleh Persamaan (2.7)

$$\mu[x] = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}$$
(2.7)



Gambar 2.9 Fungsi keanggotaan generalized bell

2.5 Charging Mode

Dalam *charging mode*, umumnya baterai diisi dengan metode *three stage charging* ^[3]. Metode ini sangat cocok untuk baterai jenis lead acid. Adapun langkah dari metode tersebut adalah:



Gambar 2.10 Diagram 3 fasa charging

1. Fase *bulk* (*constant current*): baterai akan di cas sesuai dengan tegangan *setup* (*bulk* – antara 14.4 sampai 14.7 volt) dan arus diambil secara maksimum dari panel surya.
Pada saat baterai sudah pada tegangan *setup* (*bulk*) dimulailah fase *absorption*.

- 2. Fase *absorption* (constant voltage): pada fase ini, tegangan baterai akan dijaga sesuai dengan tegangan *bulk*, sampai *solar charge controller timer* (umumnya satu jam) tercapai, arus yang dialirkan menurun sampai tercapai pada kapasitas baterai.
- 3. Fase *float* : baterai akan dijaga pada tegangan *float setting* (umumnya 13.4 sampai 13.7 volt). Beban yang terhubung ke baterai dapat menggunakan arus maksimum dari panel surya.

2.6 Modified Particle Swarm Optimization

PSO merupakan metode optimasi artificial intelligence yang mengadopsi perilaku sosial kawanan burung atau ikan. Perilaku sosial dari organisme tersebut baik individu maupun kawanan (swarm) dijadikan sebagai dasar dalam merancang algoritma PSO. Setiap solusi dapat dianggap sebagai partikel atau seekor burung. Burung akan mencari makanannya melalui usahanya sendiri dan kerja sama sosial dengan kawanannya. Algoritma PSO pertama kali diperkenalkan oleh R. Eberhard dan J. Kennedy pada tahun 1995. Gambar 2.11 merupakan algoritma standar dari *Particle Swarm Optimization*.

Banyak ilmuan yang telah memodifikasi PSO baik dari kecepatan partikel maupun dari bobot inersia. Salah satunya adalah Linear Decreasing Particle Swarm Optimization (LDWPSO). Algoritma LDWPSO memodifikasi nilai dari bobot inersia PSO original. Dengan mengganti persamaan nilai bobot inersia menjadi persamaan berikut :

$$w = (w_{max} - w_{min}) \times \frac{iter_{max} - Iter}{iter_{max}} + w_{min} \qquad 2.8$$

Algoritma ini pernah dipakai oleh Alrijadjis Djoewahir untuk meningkatkan Stabilitas dan akurasi Ultrasonic Motor (2012) dan hasilnya meningkat secara signifikan.

Stand	ar algoritma Particle Swarm Organization
(1)	mulai
(2)	for setiap partikel i
(3)	inisialisasi posisi dan kecepatan
(4)	end for
(5)	while (bukan jumlah iterasi maksimum)
(6)	for tiap partikel i
(7)	tentukan fitness value ψ_i
(8)	<i>if</i> ψ_i lebih baik dari <i>pbest</i> terkini
(9)	<i>then pbest</i> fitness = ψ_i , pbest = s_i^k
(10)	<i>if ψ_i lebih baik dari gbest</i> terkini
(11)	<i>then</i> gbest fitness = ψ_i , gbest = s_i^k
(12)	end for
(13)	for tiap partikel i
(14)	hitung vi ^k
(15)	update v _i ^k
(16)	end for
(17)	update ω
(18)	end while
(19)	end

Gambar 2.11 Pseudo code *Particle Swarm Optimization* (PSO) secara umum^[9]

BAB III METODOLOGI PENELITIAN

3.1 Diagram Alir

Terdapat beberapa tahapan yang dilakukan pada penelitian. Tahapan tersebut digambarkan dalam diagram alir sebagai berikut:



Gambar 3.1 Diagram alir penelitian

Penelitian dimulai dengan perumusan masalah dari topik penelitian yang akan dilakukan. Dilanjutkan dengan studi literatur dari *hand-book*, jurnal dan laman internet dengan topik seputar solar sel, SEPIC konveter, Logika fuzzy, solar tracker, charging methode, Modified Particle swarm optimization.

Tahap penelitian dilanjutkan dengan pengambilan data primer dan data sekunder. Pengambilan data primer (langsung) diantaranya adalah suhu, iradiasi, tegangan dan arus otput pv serta tegangan open baterai. Pengambilan data sekunder (spesifikasi) PV dan baterai. Data sekunder PV diantaranya tegangan open circuit (Voc), arus short circuit (Isc), tegangan MPP (Vmpp), Arus MPP (Impp), maximum power (Pmpp) pada kondisi temperatur standar (STC). Konverter yang digunakan pada penelitian ini adalah tipe SEPIC konverter. Parameter yang dibutuhkan antara lain tegangan input konverter (Vin). Tegangan Output (Vout), arus keluaran (Iout), Frekuensi pensaklaran (Fsw), Nilai Induktor (L1/L2), nilai Capasitor (Cin/Cs/Cout), dan hambatan sekunder (R)

Penelitian dilanjutkan dengan memodelkan sistem berupa pemodelan panel PV, konvereter dan Charging. Pemodelan PV dan charging dilakukan pada perangkat lunak MATLAB 2015a. Tujuan dilakukannya pemodelan PV untuk mengetahui karakteristik modul PV pada irradiasi matahari dan suhu kerja tertentu. Pemodelan konverter dilakukan pada perangkat lunak PSIM 9.0 dan ISIS 7.9. Tujuan pemodelan konverter untuk mengetahui kinerja dari konverter yang telah dirancang.

Tahap berikutnya melakukan pengambilan data berupa data irradiasi matahari dan suhu pada panel PV two axes solar tracker dari pukul 07.00 - 17.00. Pengambilan data dilakukan selama sehari dengan kondisi cuaca surabaya cerah dan kondisi panel PV tidak terhalang oleh suatu apapun.

Tahap selanjutnya merupakan perancangan Charging dan sub system Fuzzy-MPSO berdasarkan karakteristik modul PV dan SEPIC konverter yang digunakan. Menentukan rule base pada fuzzy dengan menambahkan algoritma MPSO untuk mengoptimasi fungsi keanggotaan fuzzy. Keluaran yang diinginkan berupa duty cycle yang kemudian akan mempengaruhi tegangan yang akan masuk ke baterai. Simulasi pada perangkat lunak MATLAB 2015a digunakan untuk mengetahui performansi Charging Fuzzy-MPSO yang telah dirancang.

Tahap berikutnya adalah pembuatan SEPIC konverter yang telah disimulasikan di PSIM 9.0 dan ISIS 7.9. pada rangkaian SEPIC konverter ditambahkan sensor arus dan tegangan sebagai parameter kontrol. Setelah pembuatan hardware barulah dilakukan penyusunan software. Penyusunan software dilakukan dengan mengunakan bantuan mikrokontroler arduino. Program yang dimasukkan kedalam arduino diantaranya adalah PWM, sensor arus dan tegangan, dan kontrol Fuzzy MPSO.

Tahap Selanjutnya merupakan pengambilan data *hardware*. Tujuan dilakukannya pengambilan data *hardware* adalah untuk membandingkan hasil simulasi dan *hardware* yang telah dibuat. Selain itu juga untuk mengetahui performa dari kontrol Fuzzy MPSO yang telah dirancang.



Gambar 3.2 Diagram blok sistem secara garis besar.

Gambar 3.2 menunjukkan blok diagram sistem secara garis besar. Apabila semua pengujian sudah sesuai maka penelitian akan dilanjutkan dengan pengambilan data akhir dan analisa. Dari data dan analisa yang didapatkan selanjutnya dijadikan sebuah laporan penelitian.

3.2 Spesifikasi sistem

Modul PV yang digunakan pada penelitian adalah *type poly cristalline solar modul* dengan nominal daya sebesar 250 Wp. Spesifikasi PV ditampilkan pada tabel 3.1 yang diperoleh dari datasheet.

Model Poly cristalline Maximum Power at STC (P_{MPP}) 250 W Maximum Power Voltage at STC (V_{MPP}) 30 V Maximum Power Current at STC (I_{MPP}) 8.34 A Open Circuit Voltage at STC (V_{OC}) 36.8 V Short Circuit Current at STC (I_{SC}) 9 A -0.32 %/°C Temperature Coefficient of V_{OC} 0.06 %/°C Temperature Coefficient of ISC Series Connected Cell per Modul 60

 Tabel 3.1 : Spesifikasi Panel Surya yang digunakan

Berikut adalah kurva karakteristik Panel surya yang disimulasikan di software Matlab2015a pada kondisi *Standart Temperature Condition* (STC).





Konverter yang digunakan adalah tipe konverter SEPIC. Konverter Sepic mampu menghasilkan diatas atau dibawah tegangan input tanpa membalik kutub tegangan input.



Gambar 3.4 Rangkaian Single Ended Primary Inductance Converter

Gambar 3.4 merupakan rangkaian SEPIC konverter dengan parameter masing-masing komponen ditunjukkan pada tabel 3.2

 Tabel 3.2 parameter Komponen konverter SEPIC yang digunakan

liomnonon	Nilai	aaturaa	
komponen	Simulasi	Alat	satuan
C1	220	220 (50)	uF(V)
L1	800	760	uH
C3	10	10 (450)	uF(V)
L2	800	760	uH
D	5	40	А
C2	3300	3300 (50)	uF(V)
MOSFET	IRF540	IRF540	-
Frek.	50000	50000	Hz

Pemodelan konverter pada PSIM bertujuan untuk mengetahui karakteristik tegangan dan Arus yang melalui komponen konverter tanpa memperhitungkan spesifikasi komponen yang digunakan.



Gambar 3.6 Karakteristik Arus yang melewati (1)C1 (2)L1 (3)Cs (4) L2 (5)R



Gambar 3.7 Karakteristik tegangan input (Merah), Tegangan Output (biru)

Gambar 3.7 menunjukkan hasil respon pemodelan konverter SEPIC. Hasilnya menunjukkan bahwa ketika diberi tegangan masukan 30V, output yang dihasilkan adalah sekitar 29V dengan duty cycle 50%. Konverter SEPIC dapat menghasilkan tegangan keluaran lebih rendah maupun lebih tinggi daripada tegangan masukan. Polaritas tegangan keluaran berbanding lurus dengan tegangan masukan (negative polarity), sehingga hasilnya negatif. Hal ini menunjukkan pemodelan sudah cukup sesuai dengan Teori yang dijelaskan.

3.3 Pengambilan Data Primer

Pengambilan data primer meliputi data Irradiasi, Suhu panel, Tegangan dan Arus. Dengan menggunakan baterai 12Ah sebagai beban yang terlebih dahulu terhubung ke *Solar Charge Controller* (SCC). Pengambilan data dilakukan pada tanggal 17 April 2018 bertempat di lantai 3 gedung S2 Departemen Teknik Fisika ITS mulai jam 6.00 hingga 17.00 WIB. Lokasi ini terhindar dari bayangan (*Shading*) yang dapat menghalangi Irradiasi ke permukaan Panel Surya. Pengukuran Irradiasi dilakukan dengan menggunakan Termometer infra merah. Pengukuran arus dan tegangan dilakukan dengan multimeter digital.

Tabel 3.3 menampilkan hasil pengambilan data irradiasi matahari dan suhu panel selama sehari (13 jam operasi). Diperoleh irradiasi dan suhu panel tertinggi pada pukul 11.30 dengan nilai 1011,333 W/m² dan 61,667 °C. Seperti yang sudah dijelaskan di sub bab 2.1 pada tabel 3.3 tegangan PV mulai menurun pada jam 10.00 hingga jam 14.00 dikarenakan panas yang diterima oleh permukaan PV menyebabkan penurunan nilai tegangan output PV. Sedangkan nilai arus pada jam yang sama mengalami peningkatan diakibatkan irradiasi yang diterima permukaan PV.

<u>د</u>	Surya						
Waktu	Irradiasi	Suhu	Tegangan	Arus			
6:00	31,5	26,867	28,62	0,251			
6:30	59,367g	27,667	29,733	0,53			
7:00	194,9	31,8	30,86	1,524			
7:30	343,7	50,667	30,667	3,093			
8:00	419,5	41,2	30,6	3,364			
8:30	263,067	41,533	29,43	2,2			
9:00	709,933	47,6	30,55	5,43			
9:30	227,533	41,6	29,92	1,93			
10:00	923,933	55	30,25	6,267			
10:30	946,4	57,267	29,576	5,667			
11:00	1000,967	59,133	29,66	6,633			
11:30	1011,333	61,667	29,25	5,76			
12:00	990,333	58,267	29,333	5,813			
12:30	938,933	59,533	28,86	6,826			
13:00	912,167	55,867	29,336	6,75			
13:30	793,767	57,533	29,133	6,6			
14:00	706,933	58,2	29,62	4,446			
14:30	518,8	48,2	30,16	3,822			
15:00	74,1	36,4	28,65	0,619			
15:30	67,467	33,533	28,8	0,562			
16:00	56,333	33,8	28,41	0,445			
16:30	33,867	32,667	27,51	0,266			
17:00	13	30,4	25,76	0,1			

Tabel 3.3Data Irradiasi, Suhu, Arus dan Tegangan Panel
Surya

3.4 Perancangan kontrol Fuzzy-MPSO

Terdapat empat elemen penyusun kontrol fuzzy yaitu fuzzifikasi, Inference system, rule base dan defuzzifikasi.

Kontrol fuzzy digunakan untuk men-*tracking* output dari konverter SEPIC agar sesuai dengan tegangan pengisian baterai. Input yang diberikan pada kontroler berupa Error tegangan (error) dan Delta Eror tegangan (derror), sedangkan output yang dihasilkan berupa duty cycle (D). Gambar 3.8 merupakan diagram blok sistem kontrol Fuzzy-MPSO. Set point yang digunakan merupakan nilai tegangan charging yaitu 14.55. Eror merupakan selisih tegangan set point dan intput. Delta error adalah selisih error saat ini dengan eror sebelumnya.



Gambar 3.8 Diagram blok kontrol Fuzzy MPSO

Membership function (fungsi keanggotaan) yang digunakan adalah tipe segitiga dan trapesium dengan 3 MF segitiga dan 2 MF trapesium.Fungsi keanggotaan segitiga dan trapesium digunakan karena fungsi ini dapat lebih mudah mikrokontroller dibaca oleh dalam menentukan batas membership functiion. Fungsi trapesium untuk mengatasi error dan delta error input yang besar akibat error pembacaan sensor tegangan sehingga Input error dan delta error masih terbaca dalam rentang fungsi keanggotaan.



Gambar 3.9 Fungsi keanggotaan input Eror dan delta eror Tegangan

Gambar 3.9 menunjukkan Fungsi keanggotaan eror (error) dan delta eror (derror) yang akan menjadi input pada kontrol Fuzzy. Bilangan input *crisp* yang dinyatakan dalam fungsi keanggotaan fuzzy akan diolah pada *inference engine* yang memiliki prinsip sebab akibat (IF..THEN). sehingga dapat menghasilkan output berupa nilai duty cycle.



Gambar 3.10 Fungsi keanggotaan output Duty Cycle

Gambar 3.10 menunjukkan fungsi keanggotaan duty cycle yang telah diatur berdasarkan parameter kerja konverter SEPIC yang telah dirancang.

Pada *inference engine* terdapat *Rule base* yang berfungsi untuk mengatur keluaran *Duty cycle* agar sesuai dengan masukan yang diterima oleh sistem. *Rule Base* berisi sekumpulan aturan sebab-akibat (IF.. THEN..) yang bertugas mengambil keputusan akhir logika fuzzy. *Rule Base* dirancang sebanyak 25 aturan (*rule*) sesuai dengan input fungsi keanggotaan (*Membership Function*).

Tabel 3.4 menunjukkan rule base yang digunakan dalam perancangan kkontrol fuzzy. Rule yang telah dirancang menggunakan operator AND. Misal untuk rule (1) jika nilai error bagian dari fungsi keanggotaan NB dan derror bagian dari fungsi keanggotaan NB, maka nilai duty cycle akan termasuk dalam fungsi keanggotaan PB. Penjelasan singkat, IF e (NB) AND de (NB) THEN duty cycle (ZE). Demikian seterusnya dan berlaku untuk semua rule.

		derror					
		NB	N	Z	Р	PB	
	NB	PB (1)	PB (2)	P (3)	Z (4)	Z (5)	
Ē	Ν	P (6)	P (7)	Z (8)	Z (9)	Z (10)	
or	Z	Z (11)	Z (12)	Z (13)	Z (14)	Z (15)	
	Р	Z (16)	Z (17)	N (18)	N (19)	N (20)	
	PB	N (21)	N (22)	N (23)	NB (24)	NB (25)	

Tabel 3.4 Rule Base Fuzzy

Keterangan tabel 3.4:

NB : Negative Big

N : Negative

Z : Zero

P : Positive

PB : *Positive Big*

Setelah kontrol fuzzy dibuat selanjutnya melakukan optimasi fungsi keanggotaan fuzzy dengan algoritma (Modified Particle Swarm Optimization) MPSO. MPSO berperan dalam pencarian parameter batas fungsi keanggotaan terbaik dari fuzzy yang dirancang berdasarkan fungsi objektif yang telah ditetapkan yaitu Mean Square Error (MSE). Modified yang digunakan pada penelitian ini berupa modified pada nilai inersia pemberat (w). Nilai inersia pemberat dibuat linear menurun (LDWPSO) pada setiap tahap iterasi dengan mengunakan persamaan 2.8

Fungsi keanggotaan fuzzy input (error dan derror) yang akan dioptimasi adalah fungsi segitiga dan trapesium dengan masing-masing mempunya fungsi keanggotaan 5 MF. Langkah-langkah dalam melakukan optimasi fungsi keanggotaan fuzzy dengan PSO dilakukan dengan tahapan seperti gambar 3.11



Gambar 3.11 Langkah optimasi PSO

Algoritma dimulai dengan memberikan parameter awal diantaranya jumlah patikel (n), iterasi maksimal (imax), nilai inersia pemberat awal (ω_{max}), nilai inersia pemberat akhir (ω_{min}), faktor *cognitive rate* (C1), faktor *social rate* (C2). Nilai dari masing-masing parameter adala sebagai berikut: imax=10; n=10; wmax=0.7; wmin=0.3; c1=0.7; c2=0.3;

Batas-batas yang sudah ditentukan sebelumnya akan digunakan sebagai gbest awal. Nilai a adalah gbest¹, b adalah gbest², dan c adalah gbest³, seperti berikut : $Gbest^{1} = -1$ $Gbest^{2} = 0$ $\text{Gbest}^3 = 1$

Langkah berikutnya adalah menginisiasi random posisi awal (S°) untuk tiap dimensi dan juga melakukan inisialisasi nilai kecepatan awal (V°) untuk tiap partikel pada tiap-tiap dimensi

S ⁰¹	S ⁰²	S ⁰³	\mathbf{V}^{01}	\mathbf{V}^{02}	V ⁰³
-0,070	0,020	0,090	0	0	0
-0,020	-0,010	0,070	0	0	0
-0,100	-0,020	0,089	0	0	0
-0,080	0,040	0,082	0	0	0
-0,095	0,015	0,087	0	0	0
-0,092	-0,012	0,088	0	0	0
-0,076	-0,030	0,090	0	0	0
-0,088	0,030	0,079	0	0	0
-0,086	0,020	0,095	0	0	0
-0,085	0,000	0,085	0	0	0

Tabel 3.5 Inisialisasi awal posisi dan kecepatan partikel

Untuk iterasi 0, pbest untuk tiap-tiap partikel menggunakan posisi awal (s⁰) sebagai perbandingan untuk iteras-iterasi berikutnya. Langkah berikutnya adalah menghitung MSE yang dihasilkan oleh posisi partikel dengan melihat nilai μ yang dihasilkan oleh posisi partikel menggunakan batas-batas yang sudah ditentukan sebelumnya. Sebagai contoh, dihitung menggunakan partikel pertama :

$$\mu^{01} = \frac{x-a}{b-a} = \frac{0.07 - (-0.1)}{0 - (-0.1)} = 0.3$$

$$\mu^{02} = \frac{c - x}{c - b} = \frac{0.1 - (0.02)}{0.1 - (0)} = 0.8$$

$$\mu^{03} = \frac{c-x}{c-b} = \frac{0.1 - (0.09)}{0.1 - (0)} = 0.1$$

Langkah berikutnya menghitung nilai MSE (*mean square error*) yang dicapai :

MSE =
$$\frac{(\mu^{01} - y_1)^2 + (\mu^{02} - y_2)^2 + (\mu^{01} - y_1)^2}{y_1^2 + y_2^2 + y_3^2}$$
$$= \frac{(0.3 - 0)^2 + (0.8 - 1)^2 + (0.1 - 0)^2}{0^2 + 1^2 + 0^2} = 0.189$$

Proses Perhitungan MSE dilakukan pada setiap partikel, dan didapatkan hasil sebagai berikut:

μ^{01}	μ^{02}	μ^{03}	MSE
0,3	0,8	0,1	0,189
0,8	0,9	0,3	0,480
0	0,8	0,11	0,080
0,2	0,6	0,18	0,537
0,05	0,85	0,13	0,056
0,08	0,88	0,12	0,044
0,24	0,7	0,1	0,283
0,12	0,7	0,21	0,271
0,14	0,8	0,05	0,094
0,15	1	0,15	0,043

Tabel 3.6 Hasil perhitungan MSE tiap partikel

Dari perhitungan nilai MSE masing-masing partikel didapatkan bahwa nilai MSE terkecil dimiliki oleh partikel 10 dengan nilai MSE = 0,043. Sehingga kondisi terbaik yang

pernah dicapai oleh semua partikel adalah pada posisi (-0,085; 0; 0,085) dan akan dijadikan Gbest baru di iterasi selanjutnya.Langkah selanjutnya menghitung nilai inersia (ω) untuk iterasi ke-0 menggunakan persamaan 2.8

$$w = (w_{max} - w_{min}) \times \frac{iter_{max} - Iter}{iter_{max}} + w_{min}$$
$$= (0.07 - 0.03) \times \frac{10 - 0}{10} + 0.03 = 0.07$$

Langkah berikutnya adalah menghitung kecepatan (v) dan posisi (s) baru partikel untuk iterasi berikutnya menggunakan Persamaan berikut :

 $\begin{aligned} v^1 &= (\omega v^0 + c_1 r_1) \times (Pbest - s^0) + c_2 r_2 \times (Gbest - s^0) \\ s^1 &= (s^0 + v^1) \end{aligned}$

Sehingga diperoleh hasil seperti tabel 3.7 :

	1	1		1 1	
<u>S¹¹</u>	S ¹²	S ¹³	\mathbf{V}^{11}	\mathbf{V}^{12}	V ¹³
-0,092	0,006	0,097	-0,022	-0,014	0,007
-0,078	-0,003	0,092	-0,058	0,007	0,022
-0,100	-0,006	0,097	0,000	0,014	0,008
-0,094	0,011	0,095	-0,014	-0,029	0,013
-0,099	0,004	0,096	-0,004	-0,011	0,009
-0,098	-0,003	0,097	-0,006	0,009	0,009
-0,093	-0,008	0,097	-0,017	0,022	0,007
-0,097	0,008	0,094	-0,009	-0,022	0,015
-0,096	0,006	0,099	-0,010	-0,014	0,004
-0,096	0,000	0,096	-0,011	0,000	0,011

Tabel 3.7 Hasil Update posisi dan kecepatan partikel

Setelah memperbaharui nilai kecepatan (v) dan nilai posisi (s) maka dilanjutkan dengan iterasi selanjutnya yaitu

iterasi ke-1. Cara yang sama dilakukan dengan terlebih dahulu menghitung nilai MSE seperti pada iterasi ke 0 untuk iterasi selanjutnya. Sehingga diperoleh hasil dari optimasi fungsi keanggotaan Input eror, deror dan output duty cycle seperti gambar berikut:



Gambar 3.12 fungsi keanggotaan fuzzy-LDWPSO

Gambar 3.12 merupakan fungsi keanggotaan fuzzy yang telah di optimisasi oleh LDWPSO (*Linear Decreasing Weight*

Particle Swarm Optimization) fungsi keanggotaan ini yang akan di inputkan ke mikrokontroller Arduino.

3.5 Perancangan Hardware

Dari parameter konverter SEPIC yang telah dihitung sebelumnya dibuat rangkaian hardware dengan parameter nilai masing –masing komponen. Gambar 3.2 menampilakan diagram alir hardware secara keseluruhan :

1. Konverter SEPIC



Gambar 3.13 Hardware konverter SEPIC

Seperti yang sudah dijelaskan pada sub bab 2.3 bahwa konverter SEPIC terdiri atas Dua buah induktor, dua buah

kapasitor, sebuah dioda dan MOSFET sebagai saklar dan rangkaian pengontrol (*Driver*) untuk mengatur MOSFET. Nilai dari masing masing komponen dijelaskan sebagai berikut:

a. Induktor

Induktor dirangkai dari inti serbuk besi dengan diameter luar 2,5 cm, diameter dalam 1,35 cm, tinggi 1,05 cm dan kabel tembaga dengan diameter 0,165 cm dengan jumlah lilitan 12. Induktor memiliki nilai 0,76mH yang terukur pada LCR meter.



Gambar 3.14 Pengukuran nilai Induktor

b. Capasitor

Capasitor Cs pada rangkaian konverter SEPIC berdasarkan perhitungan haruslah bernilai 7,633uF 206,466 V. Pada rangkaian dipasang 10uF 250 V. Capasitor Cout pada perhitungan dibutuhkan 490,672uF sedangkan Pada rangkaian dipasang 3300uF. Capasitor Cout berfungsi untuk meminimalisir osilasi (*ripple*) arus yang masuk pada baterai.

c. MOSFET

Berdasarkan hasil perhitungan mosfet yang dibutuhkan minimal memiliki peak tegangan 51,53 Volt dan peak arus 11,056 Ampere. Pada rangkaian digunakan mosfet IRF540 yang memiliki peak tegangan 100Volt dan peak arus 28 Ampere. Mosfet tipe IRF memiliki tegangan operaional gate sebesar $\pm 10V$. Sehingga sinyal PWM 5 Volt dari arduino harus dikuatkan menjadi 10 Volt. Penguatan menggunakan TLP250 module ouptocoupler agar mendapatkan sinyal PWM dengan tegangan input yang digunakan berasal dari aki 12V. Berikut rangkaian dan sinyal yang dihasilkan setelah penguatan oleh TLP250



Gambar 3.15 Rangkaian penguat sinyal PWM TLP250



Gambar 3.16 Sinyal PWM dari arduino



Gambar 3.17 Sinyal PWM yang telah dikuatkan oleh TLP250

Pada gambar 3.16 signal *duty cycle* berasal dari output PWM arduino dengan tegangan sebesar 5V. Pada gambar 3.17 signal tegangan *duty cycle* yang telah dikuatkan oleh optocoupler TLP250. Signal PWM ini dibutuhkan oleh gate mosfet agar dapat bekerja secara maksimal.

d. Resistor



Gambar 3.18 : Pengukuran resistor

Resistor digunakan sebagai *sensing element* arus yang keluar dari output konverter SEPIC. Berdasarkan perhitungan resistor yang digunakan bernilai 10 Ohm sedangkan pada rangkaian digunakan resistor 4,7 ohm dan 6,8 ohm masing-

masing memiliki spesifikasi daya 50 watt yang disusun seri sehingga nilai yang terukur pada Ohm meter bernilai 11,60hm.

2. Mikrokontroller

Mikrokontroller yang digunakan pada penelitian ini adalah jenis arduino uno. Arduino Uno adalah board mikrokontroler berbasis ATmega328. Arduino uno memiliki 14 pin digital input / output (dimana 6 dapat digunakan sebagai output PWM), 6 input analog, resonator keramik 16 MHz.



Gambar 3.19 Arduino Uno

Pin yang digunakan pada penelitian ini adalah pin A1 sebagai pembaca nilai ADC tegangan yang diukur oleh sensor tegangan dan pin PWM 9 yang digunakan sebagai pengatur nilai *duty cycle* yang telah diputuskan oleh kontroller.

3. Sensor Tegangan

Sensor tegangan menggunakan *modul voltage sensor* sebagai pengukuran tegangan keluaran konverter. Sensor ini memiliki range pembacaan tegangan DC sebesar 0-25Vdc sedangkan resolusi pembacaan sensor sebsar 0,00489 V.



Gambar 3.20 Sensor Tegangan

Sinyal tegangan yang diukur oleh sensor ini yang selanjutnya diproses oleh mikrokontroller menjadi nilai *error* dan delta error tegangan. *Error* dan *delta error* tegangan selanjutnya diproses oleh mikrokontroller untuk menghasilkan nilai *duty cycle* yang diharapkan.

Halaman ini sengaja dikosongkan

BAB IV ANALISA DATA DAN PEMBAHASAN

4.1 Simulasi konverter SEPIC pada ISIS

Simulasi pada ISIS ini dilakukan untuk mengetahui rangkaian konverter yang telah di desain dapat bekerja dengan baik. Hasil dari simulasi kemudian diimplementasikan pada *hardware* dan dilakukan pengambilan data. Data hasil simulasi dan alat ditampilkan pada tabel 4.1 :



Gambar 4.1 : Setup Simulasi SEPIC pada software ISIS9.0



Gambar 4.2 : Setup alat SEPIC

Gambar 4.1 menampilakan setup alat yang disimulasikan pada proteus. Simulasi pada proteus ini yang nantinya akan dibandingkan dengan pengambilan data hardware untuk mengetahui kinerja konverter sudah baik.

		Alat			Sin	nulasi I	SIS
Duty	Vin	Iin	Vout	Iout	Iin	Vout	Iout
10	10	0,041	0,5	0,033	0,01	0,66	0,05
	15	0,074	0,8	0,073	0,03	1,2	0,1
20	10	0,167	2,2	0,138	0,06	2,01	0,17
	15	0,239	3,8	0,237	0,06	3,32	0,29
30	10	0,388	4,2	0,358	0,36	3,63	0,32
	15	0,554	6,8	0,414	0,52	5,71	0,5
40	5	0,412	2,9	0,179	0,32	2,73	0,24
	10	0,855	6,8	0,415	0,56	6,08	0,53
	12	1,195	7,9	0,464	0,6	9,42	0,82
50	5	0,745	4,2	0,26	0,8	4,2	0,37

Tabel 4.1 : data tegangan dan arus simulasi dan alat

Tabel 4.1 menunjukkan perbedaan arus dan tegangan antara alat (*real plant*) dan simulasi pada ISIS. Terdapat beberapa perbedaan pembacaan arus dan tegangan antara simulasi dan alat. Untuk lebih jelasnya Gambar 4.3



Gambar 4.3 Grafik perbedaan Tegangan dan arus simulasi dan alat

Gambar 4.3 menunjukkan tegangan output terhadap input dengan variasi duty cycle berdasarkan Tabel 4.1. Pembacaan tegangan output pada simulasi dan hardware memiliki perbedaan yang tidak terlalu signifikan. Sehingga dapat dikatakan perancangan hardware sudah sesuai.

4.2 Pengujian Karakteristik konverter SEPIC

Dilakukan uji karakteristik pada konverter SEPIC yang telah dirancang dengan beberapa variasi frekuensi *switching* dan Duty cycle untuk mengetahui karakteristik konverter.

 Tabel 4.2 Pengaruh Frekuensi switching terhadap tegangan output konverter

Vin	Frekuensi Switching					
V III	10k	20k	30k	40k	50k	60k
1	0,15	0,11	0,13	0,13	0,13	1,4
2	0,45	0,49	0,48	0,5	0,48	0,49
3	0,79	0,86	0,83	0,88	0,84	0,86
4	1,16	1,28	1,2	1,26	1,23	1,21
5	1,56	1,6	1,6	1,64	1,64	1,6
6	2,5	2,03	1,96	2,04	2,02	1,96
7	2,87	2,37	2,35	2,38	2,42	2,32
8	NA	2,73	2,74	2,8	2,82	2,81
9	NA	3,11	3,17	3,17	3,21	3,2
10	NA	3,52	3,53	3,56	3,62	3,6
11	NA	3,94	3,89	3,94	4,01	3,95
12	NA	4,5	4,35	4,33	4,43	4,45
13	NA	NA	4,75	4,7	4,84	4,8
14	NA	NA	5,18	5,1	5,23	5,25
15	NA	NA	5,6	5,53	5,61	5,64
16	NA	NA	6	6,02	6,04	6

Pengujian variasi frekuensi dilakukan untuk mengetahui pengaruh frekuensi pensaklaran (*switching*) yang digunakan terhadap tegangan output. Frekuensi Pensaklaran pengujian dilakukan dengan rentang 10.000Hz-60.000Hz. Pengujian dilakukan dengan sumber tegangan DC berupa power generator (16V/2A), beban konverter 6,8 Ohm dan duty cycle 30% . Hasil pengujian variasi frekuensi ditampilakan pada tabel 4.2

Tabel 4.2 menunujukkan pegaruh frekuensi switching terhadap keluaran Output konverter SEPIC. NA (Not Available) merupakan kondisi dimana generator tidak dapat menghasilkan tegangan input. Terlihat frekuensi *switching* tidak terlalu berpengaruh terhadap tegangan output yang dihasilkan konverter. Konverter mulai bekerja dengan baik ketika diberikan frekuensi switching sebesar 30 KHz.

Vin			Duty		
V III	10%	20%	30%	40%	50%
1	0,01	0,03	0,12	0,31	0,32
2	0,03	0,2	0,49	0,87	0,92
3	0,09	0,41	0,83	1,46	1,52
4	0,16	0,62	1,17	2,03	2,09
5	0,26	0,85	1,56	2,61	2,64
6	0,35	1,11	1,93	3,21	3,22
7	0,45	1,38	2,3	3,73	3,81
8	0,55	1,56	2,67	4,25	4,37
9	0,64	1,8	3,04	4,89	5,02
10	0,74	2	3,45	5,42	5,63
11	0,84	2,24	3,8	6,05	6,21
12	0,94	2,5	4,13	6,65	6,75

Tabel 4.3Pengaruh *Duty cycle* terhadap tegangan output
konverter

Pengujian variasi *duty cycle* dilakukan untuk mengetahui pengaruh duty cycle terhadap tegangan output yang dihasilkan oleh konverter. Pengujian dilakukan dengan generator DC (16V/2A), beban resistor 6,8 Ohm dan frekuensi *switching* 50 KHz. Hasil pengujian variasi *duty cycle* ditampilkan pada tabel 4.3 dan gambar 4.4 :



Gambar 4.4 Pengaruh Pengaruh Duty cycle terhadap tegangan output konverter

Gambar 4.4 menunjukkan respon tegangan output konverter SEPIC dengan variasi duty cycle. Semakin besar nilai duty cycle semakin besar nilai tegangan output. Penigkatan tegangan ouptut akibat penigkatan nilai *duty cycle* terlihat konstan. Hal ini menunjukkan konverter dapat merespon dengan baik perubahan nilai *duty cycle*.

4.3 Simulasi kontrol Penjejakan Set Point Fuzzy dan Fuzzy-LDWPSO

Simulasi penjejakan ini dilakukan untuk mengetahui perbandingan unjuk kerja kontroler Fuzzy dan Fuzzy-LDWPSO yang telah dirancang dalam men-tracking set point charging (14.55 V). Simulasi dilakukan pada kondisi STC, yaitu kondisi irradiasi sebesar 1000 W/m² dan Suhu 25 0 C.

Gambar 4.5 garis merah menunjukkan tegangan output konverter SEPIC dengan menggunakan kontrol fuzzy, biru menunjukkan tegangan output dengan kontrol Fuzzy-LDWPSO dan garis hijau tegangan set point yaitu 14.55 V. Terlihat bahwa riple tegangan yang dialami kontrol Fuzzy berada di angka 15,8V sampai 19,8V sedangkan fuzzy-LDWPSO memiliki riple tegangan output pada 14,7 V sampai 16,3V.

ED III DO		
Parameter	Fuzzy	Fuzzy LDWPSO
Rise Time	0,0027 detik	0,0021
Settling Time	0,0189 detik	0,0135
Maximum Overshoot	43,64% (20,9 V)	27,5% (18,7 V)
Peak Time	0,008 detik	0,0056
Error Steady state (0,11	36,08%	8,59%
detik)	(19,8 V)	(15,8 V)

 Tabel 4.4
 Perbandingan parameter kontrol fuzzy dan fuzzy

 LDWPSO

Pada Tabel 4.4 terlihat kontrol Fuzzy LDWPSO memiliki respon yang lebih cepat dibandingkan kontrol Fuzzy. Hal ini dapat dilihat berdasarkan penurunan nilai *rise time, settling time,* dan *peak time.* Nilai *maximum overshoot* dan *error steady state* kontrol fuzzy LDWPSO lebih kecil dibandingkan dengan fuzzy. Dari data diatas fuzzy-LDWPSO lebih baik dalam men-*tracking* tegangan charging dengan menurunkan *maximum Overshoot* sebesar 16,14 % dan menurunkan error steady state sebesar 27,49 % dari kontrol fuzzy.



Gambar 4.5 Tracking tegangan 14.55 volt pada kondisi STC

4.4 Simulasi Uji Tracking Fuzzy dan Fuzzy-LDWPSO dengan Variasi Kondisi Klimatik

Uji ini dilakukan untuk mengetahui seberapa baik perbandingan kontrol fuzzy dan fuzzy-LDWPSO dalam men*tracking* tegangan set point dengan variasi kondisi suhu dan irradiasi. Variasi dan suhu irradiasi yang digunakan berdasarkan hasil dari pengambilan data sekunder irradiasi dan suhu tanggal 17 april 2018 (Tabel 3.3).



Gambar 4.6 Variasi data Irradiasi (W/m²)

Gambar 4.6 menunjukkan hasil pengukuran irradiasi dalam W/m^2 selama 10,5 jam mulai dari jam 6.00 WIB sampai 16.30 WIB. Tampak pada grafik irradiasi terbesar terjadi pada jam 11.30 WIB.



Gambar 4.7 Variasi data suhu (°C)

Gambar 4.7 menampilkan suhu permukaan panel yang terukur dalam °C selama 10,5 jam. Suhu panel tertinggi dicapai pada jam 11.30 dengan suhu yang tercatat pada pengukuran sebesar 61,66 °C. Dari variasi data suhu dan irradiasi diatas dilakukan uji tracking tegangan charging untuk mengetahui performansi kontrol Fuzzy dan Fuzzy LDW-PSO yang telah dirancang. Hasil dari uji tracking set point dengan variasi kondisi klimatik ditampilkan pada Gambar 4.8.

Pada Gambar 4.8 garis merah menampilkan hasil kontrol tegangan output yang dihasilkan oleh kontrol Fuzzy dan garis biru tegangan output Fuzzy-LDWPSO. *Ripple* tegangan yang dihasilkan kontol Fuzzy berada antara 14,8V sampai 15V, sedangkan *ripple* tegangan Kontrol Fuzzy LDWPSO berada di angka 14,6V sampai 14,8V .Hal ini menyatakan tegangan

pengisisan dengan menggunakan kontrol Fuzzy LDWPSO lebih mendekati set point dibandingkan dengan kontrol Fuzzy. Pada jam 15.00 Tegangan mengalami penurunan dibawah *set point* dikarenakan pada saat pengambilan data irradiasi dan suhu pada jam tersebut kondisi matahari tertutup awan tipis. Sehingga kontrol tidak dapat menjejak tegangan *set point*



Gambar 4.8 Hasil Uji *Tracking* Fuzzy (merah) dan Fuzzy-LDWPSO (biru) dengan Variasi Kondisi Klimatik
4.5 Simulasi *Charging* kontrol Fuzzy dan Fuzzy LDWPSO

Simulasi ini bertujuan untuk mengetahui arus charging konverter SEPIC yang masuk ke baterai dengan kondisi klimatik (kondisi suhu dan irradiasi satu hari penuh). Variasi kondisi klimatik yang digunakan seperti pada subbab 4.4. Pada simulasi matlab komponen elektronika penyusun konverter SEPIC diasumsikan bekerja secara ideal tanpa adanya daya yang hilang dari sistem konverter tersebut (daya disipasi). Hasil simulasi ditampilkan pada gambar 4.9.

Pada gambar 4.9 terlihat kedua kontrol memiliki ripple arus yang tinggi. Pada jam 15.00 arus tidak ada arus yang mengalir ke baterai dikarenakan cuaca yang mendung. Arus pengisisan yang masuk ke baterai pada simulasi mencapai 15 A. Pada simulasi diasumsikan semua komponen berada dalam kondisi ideal sehingga daya yang dihasilkan oleh panel surya semuanya tersimpan pada baterai.



Gambar 4.9 Ripple arus pengisisan baterai kontrol fuzzy(merah) dan Fuzzy LDWPSO (biru)

4.6 Pengujian hardware dengan kontrol Fuzzy LDWPSO

Pengujian hardware SEPIC dengan kontrol Fuzz-LDWPSO dilakukan di area parkir mobil departemen Teknik Fisika ITS tanggal 26 Mei 2018 . Pengambilan data dilakukan selama 8 jam dengan menggunakn PV dalam kondisi fix untuk menguji performansi konverter. Pengambilan data dilakukan setiap rentang 30 menit dengan kondisi cuaca cerah.

Gambar 4.10 menampilkan hasil pengambilan data arus yang masuk ke baterai. Arus pengisian mulai maksimal ketika memasuki jam 10.00 sampai 11.30 dengan peak arus tertinggi tercatat sebesar 2,56 A. Arus yang dikonsumsi oleh kontroler sebesar 0,056 A. Rata-rata arus yang masuk ke baterai selama pengisian sebesar 1,34 ampere.



Gambar 4.10 Arus pengisisan kontrol fuzzy-LDWPSO hardware

Gambar 4.10 menampilakn hasil pengambilan data arus yang masuk ke baterai. Arus pengisian mulai maksimal ketika memasuki jam 10.00 sampai 11.30 dengan peak arus terringgi tercatat sebesar 2,56 A. Arus yang dikonsumsi oleh kontroler sebesar 0,056 A. Rata-rata arus yang masuk ke baterai selama pengisian sebesar 1,34 ampere.

4.7 Pengujian Hardware dengan PV kondisi Fix dan *tracking*

Pengujian hardware untuk kondisi fix dan tracking dilakukan pada 1 Juni 2018 di area parkir mobil departemen Teknik Fisika ITS. Pengujian dilakukan dengan 2 PV berkapasitas masing-masing 250 WP dengan kondisi *fix* dan *tracking*. Tujuannya untuk mengetahui peningkatan Arus pengisian yang masuk ke baterai antara PV *Solar Tracker* dan PV *Fix* selama 1 hari. *Setup* Pengujian PV *Fix* dan *Tracker* ditampilakn pada gambar 4.11. Data hasil pengujian *Hardware* dengan PV kondisi *fix* dan *tracking* ditampilkan pada tabel 4.5.



Gambar 4.11 Set Up Pengujian PV Fix dan PV Tracking

	ARUS PENGECASAN								
JAM	SEPIC LDWPSO		SCC						
	FIX	TRACK	FIX			TRACKER			
			min	max	avg	min	max	avg	
06.30	0,08	0,132	-	-	-	-	-	-	
07.00	0,168	0,685	-	-	-	-	-	-	
07.30	0,23	0,702	0,6	1,5	1,05	0,5	1,7	1,1	
08.00	0,53	0,889	0,36	1,1	0,73	0,39	1,5	0,945	
08.30	0,75	0,905	0,74	1,4	1,07	0,72	1,6	1,16	
09.00	0,85	1,01	0,71	1,52	1,115	0,63	1,8	1,215	
09.30	1,16	1,16	0,35	1,5	0,925	0,38	1,7	1,04	
10.00	1,14	1,33	0,35	1,7	1,025	0,36	1,7	1,03	
10.30	1,15	1,29	0,39	1,6	0,995	0,4	1,5	0,95	
11.00	1,2	1,25	0,2	1,7	0,95	0,27	1,8	1,035	
11.30	1,1	1,28	0,56	1,9	1,23	0,5	1,9	1,2	
12.00	1,12	1,18	0,25	1,6	0,925	0,39	1,6	0,995	
12.30	1,12	1,18	0,34	1,6	0,97	0,39	1,6	0,995	
13.00	0,95	1,12	0,3	1,6	0,95	0,37	1,7	1,035	
13.30	0,35	1,15	0,32	0,32	0,32	0,9	0,9	0,9	
14.00	0,71	0,86	0,4	1,3	0,85	0,5	1,6	1,05	
14.30	0,51	0,81	0,5	1,3	0,9	0,46	1,4	0,93	
15.00	0,15	0,47	0,8	1	0,9	0,7	1,1	0,9	
15.30	0,13	0,4	0,6	0,9	0,75	0,58	1,1	0,84	
16.00	0,11	0,3	0,45	0,65	0,55	0,35	0,9	0,625	
16.30	0,43	0,44	0,41	0,41	0,41	0,6	0,6	0,6	
17.00	0,06	0,07	0	0	0	0	0	0	

 Tabel 4.5 : pengujian kondisi PV Fix dan Tracking

Pengambilan data juga menggunakan SCC sebagai pembanding dari konverter SEPIC yang telah dirancang. Cuaca saat pengambilan data cerah berawan.



Gambar 4.12 Perbandingan aru pengisian Fuzzy LDWPSO Fix, tracker, SCC fix dan tracker

Pada pengambilan data *hardware* tampak arus yang terukur masuk ke baterai tidak sebesar nilai pada simulasi. Seperti yang disebutkan sebelumnya hal ini dikarenakan daya disipasi (daya PV yang berubah menjadi daya lain misalnya panas). Berdasarkan spesifikasi mosfet IRF540 pada suhu 25°C daya disipasi maximum yang dapat ditanggulangi mosfet sebesar 150W. Jika IRF540 dioperasikan pada daya disipasi yang lebih besar maka akan terjadi kerusakan komponen. Rata-rata arus konsumsi yang digunakan kontroller dan motor tracker 0,52A

Arus pengisian yang masuk menggunakan SCC sangat fluktuatif sehingga dilakukan pencatatan arus maksimum dan minimum yang terbaca oleh ampere meter. Arus pengisisan yang masuk menggunakan SEPIC mengalami riple sekitar $\pm 0,01$ A yang terbaca pada ampere meter. PV dengan kondisi *tracking* menghasilkan arus pengisisan lebih besar dari PV kondisi FIX. Kenaikan arus pengisian yang dihasilkan oleh kondisi tracking menggunakan SEPIC konverter dengan Kontrol Fuzzy-LDWPSO sebesar 36,15 % dari panel kondisi fix sedangkan menggunakan SCC naik sebesar 11,61% dari panel kondisi fix.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan analisa data yang telah dilakukan pada bab sebelumnya diperoleh kesimpulan sebagai berikut:

- Kontrol fuzzy memiliki nilai RiseTime 0.0027 detik, SettlingTime 0.0189 detik, Maximum Overshoot 43.64%, PeakTime: 0.008 dan error steady state 36,08%.. Sedangkan kontrol Fuzzy-LDWPSO memiliki nilai RiseTime 0.0021 detik, SettlingTime 0.0135 detik, Maximum Overshoot 27,5%, PeakTime 0.0056 detik dan eror steady state 8,59 %.
- Rangkaian charging SEPIC menggunakan kontrol Fuzzy LDWPSO dengan kondisi panel tracking mampu meningkatkan arus pengisian baterai sebesar 36,15% dari panel kondisi fix.

5.2 Saran

Adapun saran untuk pengembangan pada penelitian selanjutnya adalah sebagai berikut:

1. Perlunya penelitian untuk mengurangi panas akibat dari daya disipasi yang sangat besar akibat dari arus input yang besar pada rangkaian konverter. Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Dhanabal R et al., "Comparison of Eficiencies of Solar Tracker systems with statical panel Single Axis Tracking System and Dual Axis Tracking System with Fixed Mount," *International Journal of Engineering and Technology*, 2013.
- [2] Po-Chen Cheng, Bo-Rei Peng, Yi-Hua Liu, Yu-Shan Cheng, and Jia-Wei Huang, "Optimization of a Fuzzy-Logic-Control-Based MPPT Algorithm Using the Particle Swarm Optimization Technique," *Energies*, 2015.
- [3] Abhiishek Chauhan, "MPPT Cntrol PV Charging System for Lead Acid Battery," National Institute of Technology, Rourkela, Orissa, 2014.
- [4] Subhransu Padhee, Umesh Chandra Pati, and Kamalakanta Mahapatra, "Design of photovoltaic MPPT based charger for lead acid batteries," *IEE*, 2016.
- [5] Jeff Falin, "Designing DC/DC converters based on SEPIC topology," *Power Management*, 2008.
- [6] Greeshma V J and Revathy Sasidharan, "Battery Charging Control using Fuzzy Logic based Controller in a Photovoltaic System ," *International Advanced Research Journal in Science, Engineering and Technology*, 2016.

- [7] A.Gaddaiah and K. Kishore Reddyasst, "Control of PV Charger System With SEPIC converter," *International of Journal Engineering Research and Application*, vol. 3, 2013.
- [8] P Cintula, P Hajek, and C Noguera, Handbook of Mathematical Fuzzy Logic. London: College Publication, 2011.
- [9] Nouaouria N, Boukadum M, and R Proulx, "A Survey and Positioning". Pattern Recognition," vol. 46, 2013.
- [10] Direktorat Jenderal Ketenaga Listrikan Kementrian Energi dan Sumber daya Mineral, "Statistik Ketenaga Listrikan 2015," Kementrian Energi dan Sumber daya Mineral, Jakarta, 2016.
- [11] Razan A. Jamous, Assem A. Tharrwat, Essam El. Seidy, and Bayoumi Ibrahim Bayoum, "modification of particle swarm optimization techniques and its application on stock market survey," *International Journal of Advanced Computer Science and Applications*, vol. 6, 2015.
- [12] David Sanz Morales, "Maximum Power Point Tracking Algorithms for Photovoltaic Applications," aalto university school of science and technology, 2010.

Lampiran A.1 Source code optimasi fuzzy LDWPSO.

```
clc;
clear;
close all;
9._____
  ------
% Optimasi Fuzzy MPPT menggunakan Linear
Decreasing Weight Particle Swarm Optmization
% case: Optimasi 5 membership function (2
trapesium & 3 segitiga)
§_____
 % step1:Menentukan parameter-parameter PSO
% jumlah partikel(n),iterasi
maks(imax), inersia awal(wmax), inersia akhir
% (wmin), konstanta pemberat(c1), konstanta
pemberat(c2)
imax=10; %iterasi maks(imax)
n = 10;
             %jumlah partikel
wmax=0.7;
             %inersia awal(wmax)
wmin=0.3; %inersia akhir(wmin)
c1=0.7;
              %konsatanta pemberat
c2=0.3;
               %konstanta pemberat
for i=1
   % step2:
% Inisialisasi batas fungsi keanggotaan yang
sudah ditentukan sebelumnya
% yang akan digunakan sebagai gbest awal
% NB MF : a=batas kiri=(tetap); b=batas
tengah(tetap); c=batas kanan=gbest1
% negative big membership function
(trapesium)
a(:,i)=0;
b(:,i)=0.25;
c(:,i)=0.28;
```

```
% NS MF : d=batas kiri=gbest2; e=batas
tengah(tetap); f=batas kanan=gbest3
% negative small - function (segitiga)
d(:,i)=0.25;
e(:,i)=0.28;
f(:,i)=0.31;
% ZE MF : g=batas kiri=gbest4; h=batas
tengah(tetap); j=batas kanan=gbest5
% zero membership function (segitiga)
q(:,i) = 0.28;
h(:,i) = 0.31;
j(:,i)=0.34;
% PS MF : k=batas kiri=gbest6; l=batas
tengah(tetap); m=batas kanan=gbest7
% positive small membership function
(segitiga)
k(:,i)=0.31;
1(:,i)=0.34;
m(:,i) = 0.37;
% PB MF : n=batas kiri=gbest8; o=batas
tengah(tetap); p=batas kanan=(tetap)
% positive big membership function
(trapesium)
o(:,i)=0.34;
p(:,i) = 0.38;
q(:,i) = 0.5;
% step3:
% Inisialisasi posisi awal (x) secara acak
untuk tiap dimensi dan juga
% kecepatan awal (v=0) untuk tiap partikel
pada tiap dimensi
% NB:
x1(:,i) = (c(:,i) - (rand(n,1) * (c(:,i) - b(:,i))));
%constraint: b<=x1<=c</pre>
v1(:,i)=zeros(n,1);
% NS:
x2(:,i) = (e(:,i) - (rand(n,1) * (e(:,i) - d(:,i))));
```

```
%constraint: d<=x2<=e</pre>
```

```
v2(:,i)=zeros(n,1);
x3(:,i) = (f(:,i) - (rand(n,1) * (f(:,i) - e(:,i))));
%constraint: e<=x3<=f</pre>
v3(:,i)=zeros(n,1);
% ZERO:
x4(:,i) = (h(:,i) - (rand(n,1)*(h(:,i) - g(:,i))));
%constraint: q<=x4<=h</pre>
v4(:,i)=zeros(n,1);
x5(:,i) = (j(:,i) - (rand(n,1) * (j(:,i) - h(:,i))));
%constraint: h<=x5<=j</pre>
v5(:,i)=zeros(n,1);
% PS:
x6(:,i) = (l(:,i) - (rand(n,1) * (l(:,i) - k(:,i))));
%constraint: k<=x6<=1</pre>
v6(:,i)=zeros(n,1);
x7(:,i) = (m(:,i) - (rand(n,1) * (m(:,i) - l(:,i))));
%constraint: l<=x7<=m</pre>
v7(:,i)=zeros(n,1);
% PB:
x8(:,i) = (p(:,i) - (rand(n,1) * (p(:,i) - o(:,i))));
%constraint: o<=x8<=p</pre>
v8(:,i)=zeros(n,1);
% step4:
% Menghitung MSE[mean square error] yang
dihasilkan oleh setiap partikel
% (per baris) yaitu dengan menghitung nilai
"myu" [derajat keanggotaan] tiap dimensi
mvul(:,i)=1;
                                                 8
myul=derajat keanggotaan PB batas kiri (a)
(tetap)
myu2(:,i)=1;
                                                 8
myu2=derajat keanggotaan PB batas tengah (b)
(tetap)
myu3(:,i)=(c(:,i)-x1(:,i))/(c(:,i)-b(:,i)); %
```

```
myu3=derajat keanggotaan PB batas kanan (c)
```

```
v1=1;
                                              8
yl=derajat keanggotaan ideal PB batas kiri
(a) adalah 1 (tetap)
y2=1;
                                              8
y2=derajat keanggotaan ideal PB batas tengah
(b) adalah 1 (tetap)
v3=0;
                                              8
y3=derajat keanggotaan ideal PB batas kanan
(c) adalah 0
myu4(:,i)=(x2(:,i)-d(:,i))/(e(:,i)-d(:,i)); %
myu4=derajat keanggotaan PS batas kiri (d)
myu5(:,i)=1;
                                              8
myu5=derajat keanggotaan PS batas tengah (e)
(tetap)
myu6(:,i)=(f(:,i)-x3(:,i))/(f(:,i)-e(:,i));
                                              8
myu6=derajat keanggotaan PS batas kanan (f)
v4=0;
                                              8
y4=derajat keanggotaan ideal PS batas kiri
(d) adalah 0
v5=1;
                                              8
y5=derajat keanggotaan ideal PS batas tengah
(e) adalah 1 (tetap)
v6=0;
                                              8
y6=derajat keanggotaan ideal PS batas kanan
(f) adalah 0
myu7(:,i)=(x4(:,i)-g(:,i))/(h(:,i)-g(:,i)); %
myu7=derajat keanggotaan ZE batas kiri (g)
myu8(:,i)=1;
                                              8
myu8=derajat keanggotaan ZE batas tengah (h)
(tetap)
myu9(:,i)=(j(:,i)-x5(:,i))/(j(:,i)-h(:,i)); %
myu9=derajat keanggotaan ZE batas kanan (j)
v7 = 0;
                                              8
y7=derajat keanggotaan ideal ZE batas kiri
(q) adalah 0
```

```
v8=1;
                                              8
y8=derajat keanggotaan ideal ZE batas tengah
(h) adalah 1 (tetap)
v9=0;
                                              8
y9=derajat keanggotaan ideal ZE batas kanan
(i) adalah 0
myu10(:,i)=(x6(:,i)-k(:,i))/(l(:,i)-k(:,i));
% myu10=derajat keanggotaan PS batas kiri (k)
myu11(:,i)=1;
% myull=derajat keanggotaan PS batas tengah
(1) (tetap)
myu12(:,i) = (m(:,i) - x7(:,i)) / (m(:,i) - 1(:,i));
% myu12=derajat keanggotaan PS batas kanan
(m)
v10=0;
% y10=derajat keanggotaan ideal PS batas kiri
(k) adalah 0
y11=1;
% y11=derajat keanggotaan ideal PS batas
tengah (1) adalah 1 (tetap)
v12=0;
% y12=derajat keanggotaan ideal PS batas
kanan (m) adalah 0
myu13(:,i) = (x8(:,i) - o(:,i)) / (p(:,i) - o(:,i));
% myu13=derajat keanggotaan PB batas kiri (o)
myu14(:,i)=1;
% myu14=derajat keanggotaan PB batas tengah
(p) (tetap)
myu15(:,i)=1;
% myu15=derajat keanggotaan PB batas kanan
(q) (tetap)
v13=0;
% y13=derajat keanggotaan ideal PB batas kiri
(o) adalah 0
v14=1;
% y14=derajat keanggotaan ideal PB batas
tengah (p) adalah 1 (tetap)
```

```
66
```

```
v15=1;
% y15=derajat keanggotaan ideal PB batas
kanan (q) adalah 1 (tetap)
mse(:,i) = (((y1-myu1(:,i)).^2) + ((y2-
myu2(:,i)).^2)+((y3-myu3(:,i)).^2)+((y4-
myu4(:,i)).^2)+((y5-myu5(:,i)).^2)+...
    ((y6-myu6(:,i)).^2)+((y7-
myu7(:,i)).^2)+((y8-myu8(:,i)).^2)+((y9-
myu9(:,i)).^2)+((y10-myu10(:,i)).^2)+((y11-
myu11(:,i)).^2)+...
    ((y12-myu12(:,i)).^2)+((y13-
myu13(:,i)).^2)+((y14-myu14(:,i)).^2)+((y15-
myu15(:,i)).^2))/15;
% step5:
% Menghitung Inersia, fungsi inersia adalah
untuk mengurangi kecepatan
% perhitungan nilai minimum fungsi obyektif
pada iterasi berikutnya supaya
% lebih teliti sehingga tidak ada nilai yang
terlewatkan
w(:,i)=wmin-((wmax-wmin)/imax)*(i-1);
% step6:Membangkitkan nilai parameter r1 dan
r2 secara acak
r1(:,i)=rand();
r2(:,i)=rand();
% step7:Menghitung kecepatan terbaru partikel
(update kecepatan)
pbest1(:,i)=x1(:,i);
gbest1(:,i)=c(:,i);
v1(:,i+1) = (w(:,i) * v1(:,i)) + (c1*r1(:,i)) * (pbest)
1(:,i)-x1(:,i))+(c2*r2(:,i)*(gbest1(:,i)-
x1(:,i)));
pbest2(:,i) = x2(:,i);
gbest2(:,i)=d(:,i);
```

```
v2(:,i+1) = (w(:,i)*v2(:,i)) + (c1*r1(:,i)*(pbest
2(:,i)-x2(:,i)) + (c2*r2(:,i)*(gbest2(:,i)-
x2(:,i)));
pbest3(:,i)=x3(:,i);
gbest3(:,i)=f(:,i);
v3(:,i+1) = (w(:,i) * v3(:,i)) + (c1*r1(:,i) * (pbest))
3(:,i)-x3(:,i)) + (c2*r2(:,i)*(gbest3(:,i)-
x3(:,i)));
pbest4(:,i) = x4(:,i);
gbest4(:,i) = q(:,i);
v4(:,i+1) = (w(:,i) * v4(:,i)) + (c1*r1(:,i) * (pbest))
4(:,i) - x4(:,i)) + (c2*r2(:,i)*(gbest4(:,i) - 
x4(:,i)));
pbest5(:,i)=x5(:,i);
```

```
gbest5(:,i)=j(:,i);
v5(:,i+1) = (w(:,i) * v5(:,i)) + (c1*r1(:,i) * (pbest)
5(:,i)-x5(:,i))+(c2*r2(:,i)*(gbest5(:,i)-
x5(:,i)));
```

```
pbest6(:,i)=x6(:,i);
gbest6(:,i) = k(:,i);
v6(:,i+1) = (w(:,i) * v6(:,i)) + (c1*r1(:,i) * (pbest)
 6(:,i)-x6(:,i)) + (c2*r2(:,i)*(gbest6(:,i)-
x6(:,i)));
pbest7(:,i) = x7(:,i);
gbest7(:,i)=m(:,i);
v7(:,i+1) = (w(:,i) * v7(:,i)) + (c1*r1(:,i) * (pbest))
7(:,i) - x7(:,i)) + (c2*r2(:,i)*(abest7(:,i) - x7(:,i))) + (c2*r2(:,i)) + (c2*r
x7(:,i)));
```

```
pbest8(:,i)=x8(:,i);
gbest8(:,i)=o(:,i);
v8(:,i+1) = (w(:,i) * v8(:,i)) + (c1*r1(:,i) * (pbest))
8(:,i)-x8(:,i))+(c2*r2(:,i)*(gbest8(:,i)-
x8(:,i)));
```

% step8:Menghitung posisi terbaru partikel (update posisi)

```
x1(:,i+1)=x1(:,i)+v1(:,i+1);
x2(:,i+1)=x2(:,i)+v2(:,i+1);
x3(:,i+1) = x3(:,i) + v3(:,i+1);
x4(:,i+1)=x4(:,i)+v4(:,i+1);
x5(:,i+1)=x5(:,i)+v5(:,i+1);
x6(:,i+1)=x6(:,i)+v6(:,i+1);
x7(:,i+1) = x7(:,i) + v7(:,i+1);
x8(:,i+1)=x8(:,i)+v8(:,i+1);
% step9:Mencari batas baru a dan c
berdasarkan nilai minimum MSE;
[value(:,i),index(:,i)]=min(mse(:,i));
a(:,i+1)=a(:,i);
b(:,i+1)=b(:,1);
c(:,i+1)=pbest1(index(:,i));
d(:,i+1)=pbest2(index(:,i));
e(:,i+1)=e(:,1);
f(:,i+1) = pbest3(index(:,i));
g(:,i+1)=pbest4(index(:,i));
h(:,i+1)=h(:,1);
j(:,i+1)=pbest5(index(:,i));
k(:,i+1)=pbest6(index(:,i));
l(:,i+1)=l(:,1);
m(:,i+1)=pbest7(index(:,i));
o(:,i+1)=pbest8(index(:,i));
p(:,i+1) = p(:,1);
q(:,i+1) = q(:,1);
```

end

for i=2:imax

```
myu1(:,i)=1;
% myul=derajat keanggotaan PB batas kiri (a)
(tetap)
myu2(:,i)=1;
                                              8
myu2=derajat keanggotaan PB batas tengah (b)
(tetap)
myu3(:,i)=(c(:,i)-x1(:,i))/(c(:,i)-b(:,i)); %
myu3=derajat keanggotaan PB batas kanan (c)
v1=1;
                                              8
yl=derajat keanggotaan ideal PB batas kiri
(a) adalah 1 (tetap)
y2=1;
                                              8
y2=derajat keanggotaan ideal PB batas tengah
(b) adalah 1 (tetap)
v3=0;
                                              8
y3=derajat keanggotaan ideal PB batas kanan
(c) adalah 0
myu4(:,i)=(x2(:,i)-d(:,i))/(e(:,i)-d(:,i)); %
myu4=derajat keanggotaan PS batas kiri (d)
myu5(:,i)=1;
                                              8
myu5=derajat keanggotaan PS batas tengah (e)
(tetap)
myu6(:,i)=(f(:,i)-x3(:,i))/(f(:,i)-e(:,i)); %
myu6=derajat keanggotaan PS batas kanan (f)
v4=0;
                                              8
y4=derajat keanggotaan ideal PS batas kiri
(d) adalah 0
                                              8
y5=1;
y5=derajat keanggotaan ideal PS batas tengah
(e) adalah 1 (tetap)
                                              8
v6=0;
y6=derajat keanggotaan ideal PS batas kanan
```

```
(f) adalah 0
```

```
myu7(:,i)=(x4(:,i)-g(:,i))/(h(:,i)-g(:,i)); %
myu7=derajat keanggotaan ZE batas kiri (g)
```

```
myu8(:,i)=1;
                                              8
myu8=derajat keanggotaan ZE batas tengah (h)
(tetap)
myu9(:,i)=(j(:,i)-x5(:,i))/(j(:,i)-h(:,i)); %
myu9=derajat keanggotaan ZE batas kanan (j)
v7=0;
                                              8
y7=derajat keanggotaan ideal ZE batas kiri
(g) adalah 0
v8=1;
                                              8
y8=derajat keanggotaan ideal ZE batas tengah
(h) adalah 1 (tetap)
v9=0;
                                              0
y9=derajat keanggotaan ideal ZE batas kanan
(j) adalah 0
myu10(:,i) = (x6(:,i)-k(:,i)) / (l(:,i)-k(:,i));
% myu10=derajat keanggotaan PS batas kiri (k)
myu11(:,i)=1;
% myull=derajat keanggotaan PS batas tengah
(1) (tetap)
myu12(:,i) = (m(:,i) - x7(:,i)) / (m(:,i) - 1(:,i));
% myu12=derajat keanggotaan PS batas kanan
(m)
v10=0;
% y10=derajat keanggotaan ideal PS batas kiri
(k) adalah 0
v11=1;
% y11=derajat keanggotaan ideal PS batas
tengah (1) adalah 1 (tetap)
v12=0;
% y12=derajat keanggotaan ideal PS batas
kanan (m) adalah 0
myu13(:,i)=(x8(:,i)-o(:,i))/(p(:,i)-o(:,i));
% myu13=derajat keanggotaan PB batas kiri (o)
```

% myu14=derajat keanggotaan PB batas tengah

```
(p) (tetap)
```

myu14(:,i)=1;

```
myu15(:,i)=1;
% myu15=derajat keanggotaan PB batas kanan
(q) (tetap)
y13=0;
% y13=derajat keanggotaan ideal PB batas kiri
(o) adalah 0
v14=1;
% y14=derajat keanggotaan ideal PB batas
tengah (p) adalah 1 (tetap)
v15=1;
% y15=derajat keanggotaan ideal PB batas
kanan (q) adalah 1 (tetap)
mse(:,i) = (((y1-myu1(:,i)).^2) + ((y2-myu1(:,i))).^2)
myu2(:,i)).^2)+((y3-myu3(:,i)).^2)+((y4-
myu4(:,i)).^2)+((y5-myu5(:,i)).^2)+...
    ((y6-myu6(:,i)).^2)+((y7-
myu7(:,i)).^2)+((y8-myu8(:,i)).^2)+((y9-
myu9(:,i)).^2)+((y10-myu10(:,i)).^2)+((y11-
myu11(:,i)).^2)+...
    ((y12-myu12(:,i)).^2)+((y13-
myu13(:,i)).^2)+((y14-myu14(:,i)).^2)+((y15-
myu15(:,i)).^2))/15;
w(:,i)=wmax-(((wmax-wmin)/imax)*(imax-1));
%MODIFIED W dikondisikan di sini W = (wmax-
wmin) * ((imax-i) / imax) + wmax
r1(:,i)=rand();
r2(:,i) = rand();
% cek pbest
pbest1(:,i) = ((mse(:,i) < mse(:,i-</pre>
1)).*x1(:,i))+((mse(:,i-1)<mse(:,i)).*x1(:,i-
1));
pbest2(:,i) = ((mse(:,i) < mse(:,i-</pre>
1)).*x2(:,i))+((mse(:,i-1)<mse(:,i)).*x2(:,i-
1));
```

```
72
```

```
pbest3(:,i)=((mse(:,i)<mse(:,i-</pre>
1)).*x3(:,i))+((mse(:,i-1)<mse(:,i)).*x3(:,i-
1));
pbest4(:,i)=((mse(:,i)<mse(:,i-
1)).*x4(:,i))+((mse(:,i-1)<mse(:,i)).*x4(:,i-
1));
pbest5(:,i) = ((mse(:,i) < mse(:,i-
1)).*x5(:,i))+((mse(:,i-1)<mse(:,i)).*x5(:,i-
1));
pbest6(:,i) = ((mse(:,i) < mse(:,i-</pre>
1)).*x6(:,i))+((mse(:,i-1)<mse(:,i)).*x6(:,i-
1));
pbest7(:,i)=((mse(:,i)<mse(:,i-</pre>
1)).*x7(:,i))+((mse(:,i-1)<mse(:,i)).*x7(:,i-
1));
pbest8(:,i)=((mse(:,i)<mse(:,i-</pre>
1)).*x8(:,i))+((mse(:,i-1)<mse(:,i)).*x8(:,i-
1));
gbest1(:,i)=c(:,i);
gbest2(:,i)=d(:,i);
gbest3(:,i)=f(:,i);
gbest4(:,i) = q(:,i);
gbest5(:,i)=j(:,i);
gbest6(:,i)=k(:,i);
gbest7(:,i)=m(:,i);
gbest8(:,i)=o(:,i);
v1(:,i+1) = (w(:,i)*v1(:,i)) + (c1*r1(:,i)*(pbest
1(:,i)-x1(:,i)) + (c2*r2(:,i)*(gbest1(:,i)-
x1(:,i)));
v2(:,i+1) = (w(:,i) * v2(:,i)) + (c1*r1(:,i) * (pbest))
2(:,i)-x2(:,i)) + (c2*r2(:,i)*(gbest2(:,i)-
x2(:,i)));
v3(:,i+1) = (w(:,i) * v3(:,i)) + (c1*r1(:,i) * (pbest))
3(:,i)-x3(:,i))+(c2*r2(:,i)*(gbest3(:,i)-
x3(:,i)));
```

```
v4(:,i+1) = (w(:,i) * v4(:,i)) + (c1*r1(:,i) * (pbest))
4(:,i)-x4(:,i))+(c2*r2(:,i)*(gbest4(:,i)-
x4(:,i)));
v5(:,i+1) = (w(:,i) * v5(:,i)) + (c1*r1(:,i) * (pbest))
5(:,i)-x5(:,i)) + (c2*r2(:,i)*(gbest5(:,i)-
x5(:,i)));
v6(:,i+1) = (w(:,i) * v6(:,i)) + (c1*r1(:,i) * (pbest))
6(:,i)-x6(:,i))+(c2*r2(:,i)*(gbest6(:,i)-
x6(:,i)));
v7(:,i+1) = (w(:,i) * v7(:,i)) + (c1*r1(:,i) * (pbest))
7(:,i)-x7(:,i)) + (c2*r2(:,i)*(gbest7(:,i)-
x7(:,i)));
v8(:,i+1) = (w(:,i) * v8(:,i)) + (c1*r1(:,i) * (pbest))
8(:,i)-x8(:,i))+(c2*r2(:,i)*(gbest8(:,i)-
x8(:,i)));
x1(:,i+1)=x1(:,i)+v1(:,i+1);
x2(:,i+1)=x2(:,i)+v2(:,i+1);
x3(:,i+1)=x3(:,i)+v3(:,i+1);
x4(:,i+1)=x4(:,i)+v4(:,i+1);
x5(:,i+1)=x5(:,i)+v5(:,i+1);
x6(:,i+1)=x6(:,i)+v6(:,i+1);
x7(:,i+1) = x7(:,i) + v7(:,i+1);
x8(:,i+1)=x8(:,i)+v8(:,i+1);
[value(:,i),index(:,i)]=min(mse(:,i));
a(:,i+1)=a(:,i);
b(:,i+1)=b(:,1);
c(:,i+1) = pbestl(index(:,i));
d(:,i+1)=pbest2(index(:,i));
e(:,i+1)=e(:,1);
f(:,i+1)=pbest3(index(:,i));
q(:,i+1)=pbest4(index(:,i));
h(:,i+1) = h(:,1);
j(:,i+1)=pbest5(index(:,i));
```

```
k(:,i+1)=pbest6(index(:,i));
l(:,i+1)=l(:,1);
m(:,i+1)=pbest7(index(:,i));
o(:,i+1)=pbest8(index(:,i));
p(:,i+1)=p(:,1);
q(:,i+1)=q(:,1);
z(:,i)=min (mse(:,i));
end
```

Lampiran A.2 Source code RESPON COTROL

```
A=xlsread('filename.xlsx','A1:A20001');
B=xlsread('filename.xlsx','B1:B20001');
C=xlsread('filename.xlsx','C1:C20001');
D=xlsread('filename.xlsx','D1:D20001');
plot(A,B,A,C,A,D)
xlabel('time');
ylabel('tegangan');
stepinfo(B,A)
stepinfo(C,A)
```

Hasilnya :

LDWPSO: RiseTime: 0.0021 SettlingTime: 0.0135 SettlingMin: 13.3174 SettlingMax: 18.5590 Maximum Overshoot: 18.5590 PeakTime: 0.0056

Fuzzy: RiseTime: 0.0027 SettlingTime: 0.0189 SettlingMin: 16.5092 SettlingMax: 20.4089 Maximum Overshoot: 20.4089 PeakTime: 0.0078

Lampiran A.3 Perhitungan Konveter SEPIC

 $\begin{array}{ll} V_{out} &= 14.55 \ V \\ V_{in \ min} &= 30 \ V \\ V_{in \ max} &= 37 \ V \\ V_{d} &= 0,5 \ V \\ V_{rip} &= 0,1 \ V \\ I_{out} &= 5 \ A \\ f_{sw} &= 50.000 \ Hz \\ \Delta I_{L} &= 1 \end{array}$



$$D_{\min} = \frac{V_{out} + V_d}{V_{out} + V_{in \min} + V_d} = 0,289$$

$$D_{\max} = \frac{V_{out} + V_d}{V_{out} + V_{in \min} + V_d} = 0,352$$

$$L_{\min} = \frac{10^6 \times (V_{in \min} \times D_{max})}{f_{sw} \times \Delta I_L \times I_{out} \times ((V_{out} + V_d)/V_{in \min})} = 79,9 \,\mu\text{H}$$

$$C_s = \frac{10^6 \times I_{out} \times D_{max}}{f_{sw} \times V_{in \min} \times 1000 \times 0,15} = 7,852 \,\mu\text{F}$$

$$C_{out} = \frac{10^6 \times I_{out} \times D_{max}}{f_{sw} \times V_{rip} \times 1000 \times 0,7} = 503,032 \,\mu\text{F}$$

Dimana

V _{out}	= Tegangan Output konverter (V)
$V_{in min}$	₌ Tegangan Input minimal konverter (V)
V _{in max}	= Tegangan Input maksimal konverter (V)
V _d	= Tegangan dioda (V)
V _{rip}	= Osilasi Tegangan Output (V)
I _{out}	= Arus output konverter (A)
\mathbf{f}_{sw}	= Frekuensi pensaklaran konverter (Hz)
D_{min}	= Duty cycle minimum
D _{max}	= Duty cycle maksimum
ΔI_L	= Rasio osilasi arus Induktor
Cs	= Kapasitor seri (F)
C _{out}	= Kapasitor Output (F)

Lampiran A.4 Source Code control Arduino

#include "fis_header.h"

#include <PWM.h>
float duty1;
//PWM CODE
int led =9; // the pin that the LED is attached to
int32_t frequency =50000; //frequency (in Hz)
float D= 0;

//%% module tegangan int analogInput = A1; float vout = 0.0; float vin = 0.0; float soc = 0.0; float vin0 = 0.0; float R1 = 30000.0; // float R2 = 7500.0;

```
//I2c module
int value = 0;
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,2,1,0,4,5,6,7,3,POSITIVE);
//Ubah alamat 0x3F dengan alamat i2C kamu
```

```
//fuzzyLDWPSO
```

// Number of inputs to the fuzzy inference system
const int fis_gcI = 2;
// Number of outputs to the fuzzy inference system
const int fis_gcO = 1;
// Number of rules to the fuzzy inference system
const int fis_gcR = 25;

```
FIS_TYPE g_fisInput[fis_gcl];
FIS_TYPE g_fisOutput[fis_gcO];
```

```
// Setup routine runs once when you press reset:
void setup()
{
```

```
pinMode(analogInput, INPUT);
Serial.begin(9600);
```

```
InitTimersSafe();
bool success = SetPinFrequencySafe(led, frequency);
lcd.begin (16,2);
```

```
// initialize the Analog pins for output.
// Pin mode for Output: output1
```

}

// Loop routine runs over and over again forever:

```
void loop()
{
```

```
value = analogRead(analogInput);
vout = (value * 5.0) / 1024.0; // see text
vin = vout / (R2/(R1+R2));
```

```
// Read Input: Eror
g_fisInput[0] = vin - 14.55;
// Read Input: Deror
g_fisInput[1] = g_fisInput[0] - vin0;
vin0 = g_fisInput[0];
g_fisOutput[0] = 0;
```

```
fis_evaluate();
D = g_fisOutput[0];
duty1=(D*255);
pwmWrite(led,duty1);
// Set output vlaue: output1
analogWrite(2, g_fisOutput[0]);
```

```
}
```

```
FIS_TYPE a = p[0], b = p[1], c = p[2], d = p[3];
                 FIS TYPE t1 = ((x \le c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((c != d) ? ((d = c) ? 1 : ((d \le x) ? 0 : ((d = c) ? 1 : ((d \le x) ? 0 : ((d = c) ? 1 : ((d \le x) ? 1 : ((d \le x) ? 0 : ((d = c) ? 1 : ((d \le x) ? 1 : ((d : x) ? 
x) / (d - c) : 0));
                 FIS TYPE t_2 = ((b \le x) ? 1 : ((x \le a) ? 0 : ((a \le b) ? ((x \le a) ? 0 : ((a \le b) ? ((x \le a) ? 0 : ((a \le b) ? ((x \le a) ? 0 : ((a \le b) ? ((x \le a) ? 0 : ((a \le b) ? ((x \le a) ? 0 : ((a \le b) ? ((a \le b) ?
a) / (b - a) : 0));
                 return (FIS_TYPE) min(t1, t2);
  }
// Triangular Member Function
FIS TYPE fis trimf(FIS TYPE x, FIS TYPE* p)
  {
                 FIS TYPE a = p[0], b = p[1], c = p[2];
                 FIS TYPE t1 = (x - a) / (b - a);
                 FIS TYPE t_2 = (c - x) / (c - b);
                 if ((a == b) \&\& (b == c)) return (FIS TYPE) (x == a);
                 if (a == b) return (FIS_TYPE) (t2^{*}(b \le x)^{*}(x \le c));
                 if (b == c) return (FIS TYPE) (t1*(a \le x)*(x \le b));
                 t1 = min(t1, t2);
                 return (FIS TYPE) max(t1, 0);
  }
FIS TYPE fis min(FIS TYPE a, FIS TYPE b)
  {
                 return min(a, b);
  }
FIS TYPE fis max(FIS TYPE a, FIS TYPE b)
  {
                 return max(a, b);
  }
```

80

```
FIS_TYPE fis_array_operation(FIS_TYPE *array, int size,
_FIS_ARR_OP pfnOp)
 int i;
 FIS_TYPE ret = 0;
 if (size == 0) return ret;
 if (size == 1) return array[0];
 ret = array[0];
 for (i = 1; i < size; i++)
  {
   ret = (*pfnOp)(ret, array[i]);
  }
 return ret;
}
*******
// Data for Fuzzy Inference System
******
// Pointers to the implementations of member functions
_FIS_MF fis_gMF[] =
{
 fis_trapmf, fis_trimf
};
```

// Count of member function for each Input

```
int fis_gIMFCount[] = { 5, 5 };
```

// Count of member function for each Output
int fis_gOMFCount[] = { 5 };

```
// Coefficients for the Input Member Functions
FIS_TYPE fis_gMFI0Coeff1[] = { -15, -5.25, -2, -1.02 };
FIS TYPE fis gMFI0Coeff2[] = \{ -1.84, -1, -0.257 \};
FIS TYPE fis gMFI0Coeff3[] = \{ -0.468, 0, 0.214 \};
FIS TYPE fis gMFI0Coeff4[] = \{ 0.029, 1, 1.16 \};
FIS_TYPE fis_gMFI0Coeff5[] = { 0.74, 1.86, 5, 13.7 };
FIS TYPE* fis gMFI0Coeff[] = { fis gMFI0Coeff1,
fis_gMFI0Coeff2, fis_gMFI0Coeff3, fis_gMFI0Coeff4,
fis gMFI0Coeff5 };
FIS TYPE fis gMFI1Coeff1[] = \{-0.725, -0.525, -0.4, -0.156\}
}:
FIS TYPE fis gMFI1Coeff2[] = \{ -0.336, -0.2, -0.124 \};
FIS TYPE fis gMFI1Coeff3[] = \{ -0.126, 0, 0.0772 \};
FIS_TYPE fis_gMFI1Coeff4[] = { 0.071, 0.2, 0.377 };
FIS TYPE fis gMFI1Coeff5[] = \{0.297, 0.4, 0.5, 0.715\};
FIS TYPE* fis gMFI1Coeff[] = { fis gMFI1Coeff1,
fis gMFI1Coeff2, fis gMFI1Coeff3, fis gMFI1Coeff4,
fis gMFI1Coeff5 }:
FIS_TYPE** fis_gMFICoeff[] = { fis_gMFI0Coeff,
fis gMFI1Coeff };
```

```
// Coefficients for the Output Member Functions
FIS_TYPE fis_gMFO0Coeff1[] = { 0.155, 0.195, 0.25,
0.2617 };
FIS_TYPE fis_gMFO0Coeff2[] = { 0.2537, 0.28, 0.2974 };
FIS_TYPE fis_gMFO0Coeff3[] = { 0.2829, 0.31, 0.333 };
```

```
FIS_TYPE fis_gMFO0Coeff4[] = { 0.317, 0.34, 0.3545 };
FIS_TYPE fis_gMFO0Coeff5[] = { 0.3518, 0.38, 0.45, 0.5 };
FIS_TYPE* fis_gMFO0Coeff5[] = { fis_gMFO0Coeff1,
fis_gMFO0Coeff2, fis_gMFO0Coeff3, fis_gMFO0Coeff4,
fis_gMFO0Coeff5 };
FIS_TYPE** fis_gMFOCcoeff[] = { fis_gMFO0Coeff };
```

// Input membership function set int fis_gMFI0[] = { 0, 1, 1, 1, 0 }; int fis_gMFI1[] = { 0, 1, 1, 1, 0 }; int* fis_gMFI[] = { fis_gMFI0, fis_gMFI1};

// Output membership function set int fis_gMFO0[] = { 0, 1, 1, 1, 0 }; int* fis_gMFO[] = { fis_gMFO0};

```
// Rule Inputs
int fis_gRI0[] = { 1, 1 };
int fis_gRI1[] = { 1, 2 };
int fis_gRI2[] = { 1, 2 };
int fis_gRI2[] = { 1, 3 };
int fis_gRI3[] = { 1, 4 };
int fis_gRI4[] = { 1, 5 };
int fis_gRI5[] = { 2, 1 };
```

```
int fis_gRI6[] = \{2, 2\};
int fis gRI7[] = \{2, 3\};
int fis gRI8[] = \{2, 4\};
int fis_gRI9[] = { 2, 5 };
int fis gRI10[] = \{3, 1\};
int fis_gRI11[] = { 3, 2 };
int fis_gRI12[] = { 3, 3 };
int fis gRI13[] = \{3, 4\};
int fis gRI14[] = \{3, 5\};
int fis gRI15[] = \{4, 1\};
int fis_gRI16[] = { 4, 2 };
int fis gRI17[] = \{4, 3\};
int fis_gRI18[] = \{4, 4\};
int fis gRI19[] = \{4, 5\};
int fis gRI20[] = \{ 5, 1 \};
int fis_gRI21[] = { 5, 2 };
int fis gRI22[] = \{5, 3\};
int fis gRI23[] = \{5, 4\};
int fis_gRI24[] = { 5, 5 };
int* fis gRI[] = { fis gRI0, fis gRI1, fis gRI2, fis gRI3,
fis_gRI4, fis_gRI5, fis_gRI6, fis_gRI7, fis_gRI8, fis_gRI9,
fis gRI10, fis gRI11, fis gRI12, fis gRI13, fis gRI14,
fis_gRI15, fis_gRI16, fis_gRI17, fis_gRI18, fis_gRI19,
fis_gRI20, fis_gRI21, fis_gRI22, fis_gRI23, fis_gRI24 };
```

```
// Rule Outputs
int fis_gRO0[] = { 5 };
int fis_gRO1[] = { 5 };
int fis_gRO2[] = { 4 };
int fis_gRO3[] = { 3 };
int fis_gRO4[] = { 3 };
```

```
int fis_gRO5[] = \{4\};
int fis_gRO6[] = { 4 };
int fis gRO7[] = \{3\};
int fis_gRO8[] = \{3\};
int fis gRO9[] = \{3\};
int fis_gRO10[] = { 3 };
int fis_gRO11[] = \{3\};
int fis gRO12[] = \{3\};
int fis_gRO13[] = \{3\};
int fis gRO14[] = \{3\};
int fis_gRO15[] = { 2 };
int fis gRO16[] = \{3\};
int fis_gRO17[] = { 2 };
int fis gRO18[] = \{2\};
int fis_gRO19[] = { 2 };
int fis_gRO20[] = \{2\};
int fis gRO21[] = \{2\};
int fis gRO22[] = \{2\};
int fis_gRO23[] = { 1 };
int fis gRO24[] = \{1\};
int* fis_gRO[] = { fis_gRO0, fis_gRO1, fis_gRO2, fis_gRO3,
fis gRO4, fis gRO5, fis gRO6, fis gRO7, fis gRO8,
fis_gRO9, fis_gRO10, fis_gRO11, fis_gRO12, fis_gRO13,
fis_gRO14, fis_gRO15, fis_gRO16, fis_gRO17, fis_gRO18,
fis gRO19, fis gRO20, fis gRO21, fis gRO22, fis gRO23,
fis_gRO24 };
```

// Input range Min
FIS_TYPE fis_gIMin[] = { -5, -0.5 };

// Input range Max

```
86
```

```
FIS_TYPE fis_gIMax[] = { 5, 0.5 };
```

```
// Output range Min
FIS_TYPE fis_gOMin[] = { 0.2 };
```

```
// Output range Max
FIS_TYPE fis_gOMax[] = { 0.4 };
```

```
******
// Data dependent support functions for Fuzzy Inference
System
*******************
FIS_TYPE fis_MF_out(FIS_TYPE** fuzzyRuleSet,
FIS_TYPE x, int o)
{
 FIS TYPE mfOut;
 int r;
 for (r = 0; r < fis_gcR; ++r)
 {
   int index = fis_gRO[r][o];
   if (index > 0)
   {
     index = index - 1;
     mfOut = (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOCoeff[o][index]);
   else if (index < 0)
```
```
index = -index - 1;
       mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOCoeff[o][index]);
     }
    else
     {
       mfOut = 0;
     }
    fuzzyRuleSet[0][r] = fis_min(mfOut,
fuzzyRuleSet[1][r]);
  }
  return fis_array_operation(fuzzyRuleSet[0], fis_gcR,
fis max);
}
FIS_TYPE fis_defuzz_centroid(FIS_TYPE** fuzzyRuleSet,
int o)
{
  FIS TYPE step = (fis gOMax[o] - fis gOMin[o]) /
(FIS_RESOLUSION - 1);
  FIS TYPE area = 0;
  FIS TYPE momentum = 0;
  FIS_TYPE dist, slice;
  int i;
```

// calculate the area under the curve formed by the MF outputs

```
for (i = 0; i < FIS_RESOLUSION; ++i){
    dist = fis_gOMin[o] + (step * i);
    slice = step * fis_MF_out(fuzzyRuleSet, dist, o);</pre>
```

```
area += slice;
momentum += slice*dist;
}
```

```
return ((area == 0) ? ((fis_gOMax[o] + fis_gOMin[o]) / 2) :
(momentum / area));
}
```

// Fuzzy Inference System

```
******
```

```
void fis_evaluate()
```

```
{
```

```
FIS_TYPE fuzzyInput0[] = { 0, 0, 0, 0, 0 };
FIS_TYPE fuzzyInput1[] = { 0, 0, 0, 0, 0 };
FIS_TYPE* fuzzyInput[fis_gcI] = { fuzzyInput0,
fuzzyInput1, };
FIS_TYPE fuzzyOutput0[] = { 0, 0, 0, 0, 0 };
FIS_TYPE* fuzzyOutput[fis_gcO] = { fuzzyOutput0, };
FIS_TYPE fuzzyRules[fis_gcR] = { 0 };
FIS_TYPE fuzzyFires[fis_gcR] = { 0 };
FIS_TYPE* fuzzyRuleSet[] = { fuzzyRules, fuzzyFires };
FIS_TYPE sW = 0;
```

```
// Transforming input to fuzzy Input
int i, j, r, o;
for (i = 0; i < fis_gcI; ++i)
{
    for (j = 0; j < fis_gIMFCount[i]; ++j)</pre>
```

```
{
       fuzzyInput[i][j] =
          (fis_gMF[fis_gMFI[i][j]])(g_fisInput[i],
fis_gMFICoeff[i][j]);
     ł
   }
  int index = 0;
  for (r = 0; r < fis_gcR; ++r)
   {
     if (fis_gRType[r] == 1)
     {
       fuzzyFires[r] = FIS_MAX;
       for (i = 0; i < fis gcI; ++i)
        {
          index = fis_gRI[r][i];
          if (index > 0)
             fuzzyFires[r] = fis_min(fuzzyFires[r],
fuzzyInput[i][index - 1]);
          else if (index < 0)
             fuzzyFires[r] = fis_min(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
          else
             fuzzyFires[r] = fis_min(fuzzyFires[r], 1);
        }
     }
     else
     {
       fuzzyFires[r] = FIS_MIN;
       for (i = 0; i < fis\_gcI; ++i)
        {
```

```
index = fis_gRI[r][i];
          if (index > 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r],
fuzzyInput[i][index - 1]);
          else if (index < 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
          else
            fuzzyFires[r] = fis_max(fuzzyFires[r], 0);
        }
     }
     fuzzyFires[r] = fis_gRWeight[r] * fuzzyFires[r];
     sW += fuzzyFires[r];
  }
  if (sW == 0)
  {
     for (o = 0; o < fis\_gcO; ++o)
     {
       g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) /
2);
     }
  }
  else
  {
     for (o = 0; o < fis\_gcO; ++o)
     ł
       g_fisOutput[o] = fis_defuzz_centroid(fuzzyRuleSet,
o);
     }
```

```
}
  soc=(vin-11)*100/2.5;
lcd.setCursor(0, 1); //baris kedua
lcd.print("V=");
lcd.setCursor(2, 1); //baris kedua
lcd.print(vin);
lcd.setCursor(7, 1);
lcd.print("V ");
lcd.setCursor(8, 0); //bari*s pertama
lcd.print("SOC=");
lcd.setCursor(12, 0); //baris pertama
lcd.print(soc);
lcd.setCursor(0, 0); //baris pertama
lcd.print("D=");
lcd.setCursor(2, 0);
lcd.println(D);
lcd.setCursor(6, 0); //baris pertama
lcd.print(" ");
delay(10);
}
```

Lampiran A.4 schematic dan board driver TLP250





Lampiran A.5 Simulasi kontroler Konverter SEPIC matlab SIMULINK

Gambar 6 Simulasi control charging konverter sepic MATLAB 2015a

Lampiran A.6 Simulasi Konverter Sepic PSIM 9.0



Lampiran A.6 Simulasi Konverter Sepic ISIS PROTEUS

